

8. Lema M.A. et al. Flexible dual-connectivity spectrum aggregation for decoupled uplink and downlink access in 5G heterogeneous systems. *IEEE Journal on Selected Areas in Communications*, 2016, vol. 34, no. 1, pp. 2851–2865. DOI: <https://doi.org/10.1109/JSAC.2016.2615185>.
9. Niknam S. et al. A multiband OFDMA heterogeneous network for millimeter wave 5G wireless applications. *IEEE Access*, 2016, vol. 4, pp. 5640–5648. DOI: <https://doi.org/10.1109/ACCESS.2016.2604364>.
10. Vishnevsky V., Larionov A., Frolov S. Design and scheduling in 5G stationary and mobile communication systems based on wireless millimeter-wave mesh networks. *Distributed Computer and Communication Networks*, 2014, vol. 279, pp. 11–27. DOI: https://doi.org/10.1007/978-3-319-05209-0_2.
11. Vishnevsky V.M. et al. Applying graph-theoretic approach for time-frequency resource allocation in 5G MmWave backhaul network. *Advances in Wireless and Optical Communications (RTUWO)*, 2016, pp. 221–224. DOI: <https://doi.org/10.1109/RTUWO.2016.7821888>.
12. Leland W.E. et al. On the self-similar nature of Ethernet traffic. *IEEE/ACM Transactions on Networking*, 1994, vol. 2, no. 1, pp. 1–15.
13. Tsybakov B.S. Teletraffic model based on a self-similar random process. *Radiotekhnika*, 1999, no. 5, pp. 24–31. (In Russian.)

Received 21.04.2020

УДК 621.391:621.39

АЛГОРИТМЫ УПРАВЛЕНИЯ ПОТОКОВЫМ ВИДЕОТРАФИКОМ

Лихтциндер Б.Я.

Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

E-mail: lixt@psuti.ru

Рассмотрены алгоритмы управления потоковым видеотрафиком. Анализируются особенности трафика видеокодеков, который имеет явно выраженный пачечный характер. Исследуются характеристики таких потоков и их влияние на размеры очередей и задержек в узлах телекоммуникационных сетей. Приводится обобщенная формула Хинчина – Поллачека, и показана линейная зависимость числителя указанной формулы от коэффициента загрузки. Рассмотрены основные причины задержек пакетов в очередях телекоммуникационной сети, и показано влияние этих задержек на процессы управления потоковым видеотрафиком. Предлагается алгоритм управления видеопотоком, базирующийся на измерении размеров буферной памяти в оборудовании клиента. Показано, что подобный алгоритм исключает влияние задержек из цепи обратной связи системы управления. Приводятся результаты имитационного моделирования рассматриваемых процессов.

Ключевые слова: видеотрафик, управление, задержки, очереди, битрейт, алгоритм, видеокодеки, групповые потоки, моделирование

Введение

Сегодня технология потокового вещания видео все более замещает общепринятую видеотрансляцию, основанную на протоколах RTP/UDP [1]. Технология передачи с использованием протокола TCP обладает высокой надежностью, поскольку, благодаря наличию обратной связи, восстанавливает поврежденные или утерянные во время передачи пакеты. Алгоритм управления видеопотоком предусматривает возможно более полную загрузку TCP-канала, а в случае возникновения перегрузок и появления информации об увеличении числа теряемых пакетов передатчик производит их повторное воспроизведение.

Сокращая размер «окна», в течение которого ведется передача, он уменьшает интенсивность поступления пакетов, снижая тем самым возможность возникновения перегрузки.

Одним из наиболее распространенных алгоритмов стабилизации TCP-видеопотоков является алгоритм CUBIC [2]. Управление потоком осуществляется передающей стороной. При детектировании наличия потерь пакетов передатчик понижает скорость их отправки до исчезновения потерь, а затем ступенчато повышает ее по кубической степенной зависимости до появления информации о вновь возникающих потерях.

Пропускная способность TCP-канала регулируется размером «окна перегрузки». При на-

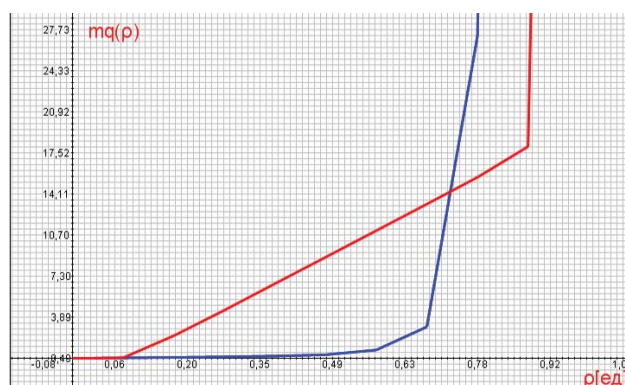


Рисунок 1. Зависимости размеров очередей от загрузки для потока UDP (нижний график) и TCP (верхний график)

личии в канале TCP-видеотрафика мы подразумеваем, что канал загружен до теоретического максимума, доступного с учетом наличия UDP-трафика. На рисунке 1 показаны зависимости размеров очередей от загрузки для потока UDP (нижний график) и TCP (верхний график). Поток UDP весьма равномерен и даже при значительных нагрузках образует небольшие очереди. Однако, имея более высокий приоритет, он вытесняет поток TCP, повышая общий коэффициент загрузки для потока TCP, оставляя ему лишь часть пропускной способности канала.

Потоки пакетов от видеокодексов

При управлении TCP-видеопотоком, в отличие от передачи трафика данных, возникают дополнительные трудности, связанные с тем, что потребление видеопакетов происходит с определенной постоянной скоростью, называемой «битрейтом».

Битрейт определяет разрешающую способность передаваемого видеосообщения. Если пропускная способность канала окажется меньше битрейта, то передача с данным уровнем качества становится невозможной, и значение битрейта необходимо уменьшить. Для этого существующие видеокодеки имеют ряд настроек, обеспечивающих ступенчатое изменение битрейта и, следовательно, качества воспроизведения.

Стандартом H.264 [3] предусмотрены «профили» кодека – набор различных функций и алгоритмов кодирования, которые абонентское устройство может уметь или не уметь выполнять. Основной список включает 10 профилей. Каждый из профилей может работать с разным качеством кодирования под названием «уровень». Всего предусмотрено 17 уровней. Таким образом, имеется 170 комбинаций «профиль – уровень».

На рисунке 2 показаны видеотрафики, генерируемые кодеком H.264-1000 (сплошные ли-

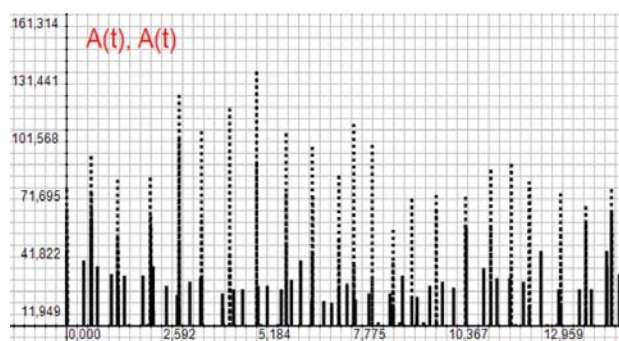


Рисунок 2. Пачечный характер трафика кодексов H.264-1000 и H.264-10000

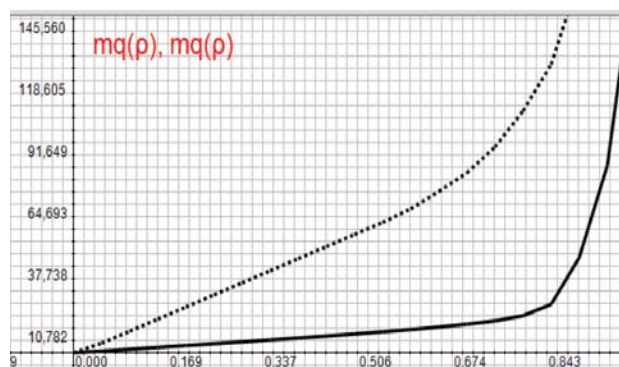


Рисунок 3. Зависимости средних значений очередей от коэффициента загрузки ρ

нии) и кодеком H.264-10000 (пунктирные линии), при одинаковых нагрузках $\rho = 0,8$. Пачки представляют совокупности пакетов, поступавших в течение времени обработки одного пакета. Оба трафика имеют явно выраженный пачечный характер. Трафик настройки H.264-1000 более равномерный и имеет среднюю интенсивность $\lambda = 72$ пакета в секунду. Настройка H.264-10000 имеет интенсивность $\lambda = 297$ пакетов в секунду и пачки, превышающие 100 пакетов. Для трафика настройки H.264-1000 особенно заметно, что интервалы между пачками непостоянны и носят случайный характер.

Большие размеры пачек у трафика H.264-10000 приводят к большим средним значениям очередей. На рисунке 3 показаны зависимости средних значений очередей от коэффициента загрузки ρ (H.264-1000 – нижняя кривая; H.264-10000 – верхняя кривая).

Пачечный характер видеотрафика позволяет получить практически линейную зависимость размеров очередей от коэффициента загрузки ρ в широком диапазоне.

Групповой неординарный пуассоновский поток

Одной из разновидностей ВМАР-потоков [4–7] является неординарный пуассоновский поток со-

бытий. В таком потоке выполняются свойства стационарности и отсутствия последствия, но не выполняется свойство ординарности. Указанные потоки предлагаются нами в качестве моделей трафика видеокодеков.

Рассмотрим пуассоновский поток независимых событий с параметром λ . Каждое событие заключается в одновременном появлении в момент t_k пачки из μ_k независимых случайно распределенных чисел заявок с распределением $P|\mu_k = k| = f_k$. Такой поток называют пуассоновским неординарным (групповым) потоком независимых событий, и он рассмотрен в [8].

Выделим интервал времени τ обработки одной заявки. Разделим промежуток времени T , в течение которого действует поток указанных событий, на N_τ последовательных интервалов τ . Пусть $m_i(\tau)$ – число событий, произошедших в течение i -го интервала времени τ .

Поскольку поток событий – пуассоновский, вероятности наступления на интервале τ ровно n событий есть

$$P|m_i(\tau) = n| = P_n(\lambda\tau) = \frac{(\lambda\tau)^n}{n!} e^{-\lambda\tau}. \quad (1)$$

Каждому из событий сопутствует появление пачки с распределением вероятностей чисел заявок f_k . Введем производящую функцию

$$f(z) = \sum_{k=0}^{\infty} f_k z^k.$$

Поскольку все μ_k взаимно независимы и одинаково распределены, появлению на интервале τ ровно $m_i(\tau)$ заявок при условии, что на указанном интервале произошло n событий, соответствует вероятность $(f_k)^n$ и производящая функция $[f(z)]^n$.

Производящая функция $G_{m(\tau)}(z)$ числа заявок на интервале τ определяется соотношением

$$\begin{aligned} G_{m(\tau)}(z) &= \sum_{n=0}^{\infty} P_n(\lambda\tau) [f(z)]^n = \\ &= \sum_{n=0}^{\infty} \frac{(\lambda\tau)^n}{n!} e^{-\lambda\tau} [f(z)]^n = \\ &= \left[\sum_{n=0}^{\infty} \frac{(\lambda\tau f(z))^n}{n!} e^{-\lambda\tau f(z)} \right] e^{\lambda\tau[f(z)-1]} = \\ &= e^{\lambda\tau[f(z)-1]}. \end{aligned} \quad (2)$$

Определим среднее число заявок на интервалах τ :

$$\begin{aligned} \overline{m(\tau)} &= \sum_{i=0}^{\infty} m_i(\tau) P[m_i(\tau)] = \frac{\partial G_{m(\tau)}(z)}{\partial z} \Big|_{z=1} = \\ &= \lambda\tau f'(z) e^{\lambda\tau[f(z)-1]} \Big|_{z=1} = \lambda\tau f'. \end{aligned}$$

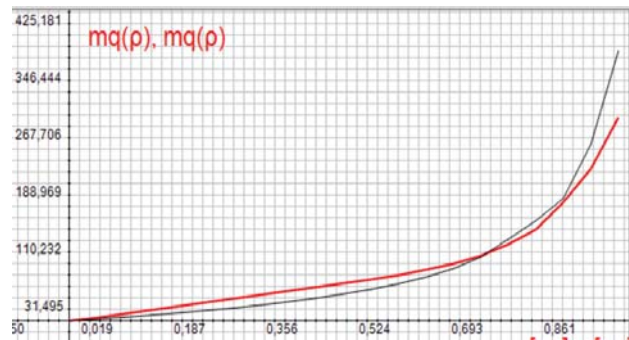


Рисунок 4. Зависимости средних размеров очередей от коэффициента загрузки ρ для трафика видеокодека H264-10 000 и для группового потока

Учитывая, что

$$f'(1) = \frac{d}{dz} \sum_{k=0}^{\infty} f_k z^k \Big|_{z=1} = \sum_{k=0}^{\infty} k f_k = \bar{k},$$

получим

$$\overline{m(\tau)} = \lambda\tau \bar{k}, \quad (3)$$

где $\bar{k}$ – среднее число заявок в пачке. Можно также показать, что дисперсию $D_m(\rho)$ числа заявок на интервалах τ есть

$$D_m(\rho) = \rho \bar{k} (1 + v_k^2), \quad (4)$$

где $\rho = \lambda\tau \bar{k} = \overline{m(\tau)}$ – общий коэффициент загрузки, а $v_k^2 = \frac{D_k}{(\bar{k})^2}$ – квадрат коэффициента вариации чисел заявок в пачках. Дисперсия $D_m(\rho)$ линейно зависит от коэффициента загрузки ρ , также, как это бывает в обычном пуассоновском потоке, где $D_m(\rho) = \rho$.

Аппроксимация

Пачечный характер реального трафика видеокодеков позволяет аппроксимировать его характеристики соответствующими характеристиками рассмотренных выше групповых пуассоновских потоков.

На рисунке 4 показаны зависимости средних размеров очередей от коэффициента загрузки для трафика видеокодека H264-10 000 и для группового потока с пуассоновским распределением чисел заявок в пачках при среднем числе заявок в пачке $\bar{k} = 100$. Мы наблюдаем весьма хорошее совпадение.

На рисунке 5 отражены изменения размеров очередей во времени для трафика видеокодека H264-10 000 и для группового потока при коэффициенте загрузки $\rho = 0,5$. Сплошной заливкой показано изменение очереди для группового пуассоновского потока.

Равномерное появление пачек пакетов видеотрафика обуславливает линейный участок зави-

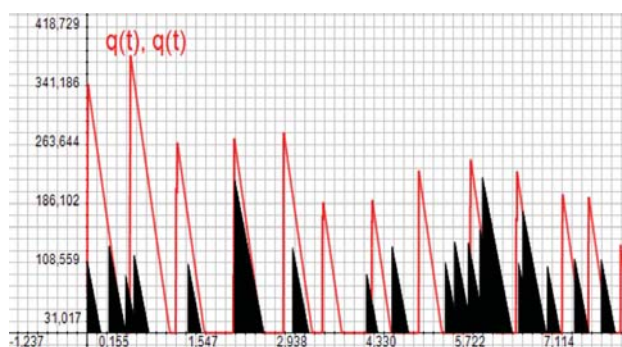


Рисунок 5. Изменения размеров очередей во времени

симости среднего размера его очереди, показанный на рисунке 4.

Устойчивость групповых пуассоновских потоков

Так же как обычные пуассоновские потоки, групповые пуассоновские потоки обладают определенной устойчивостью.

1. Поскольку они являются ординарными пуассоновскими потоками по отношению к событиям, заключающимся в возникновении пачек одновременно появляющихся пакетов, то сумма нескольких потоков таких событий также является пуассоновским потоком событий. Следовательно, сумма групповых пуассоновских потоков также остается групповым пуассоновским потоком.

2. По этой же причине сумма большого числа групповых потоков с независимыми интервалами между пачками должна сходиться к групповому пуассоновскому потоку.

3. При прохождении таким потоком быстродействующих коммутаторов в телекоммуникационной сети с дисциплиной обслуживания FIFO пришедшая почти одновременно группа пакетов будет передаваться непосредственно друг за другом. Так, что возникающие задержки могут рассматриваться как задержки пачек пакетов.

Именно по указанным причинам модель неординарного группового пуассоновского потока предлагается для анализа процессов управления потоковым видеотрафиком.

Управление переключением битрейта

Одним из наиболее часто применимых способов управления видеотрафиком является способ дискретного переключения уровня качества воспроизведения. Постоянно измеряя свою реальную пропускную способность, передатчик выбирает соответствующие значения битрейта для последующей передачи.

Весь поток разбивается на отдельные сегменты по τ с воспроизведения видео. Указанные сег-

менты могут рассматриваться как пачки пакетов. Каждый сегмент был предварительно подвергнут кодированию в один из L различных битрейтов видео. Пусть $R = \{R_1, \dots, R_L\}$ – ряд доступных битрейтов видео ($0 < R_l < R_L$ при $l \leq L$). Для каждого клиента процесс поделен на этапы загрузки сегментов $i = 1, 2, \dots$. Обозначим через ϑ_i время от начала передачи передатчиком очередного i -го сегмента до получения подтверждения о его правильном приеме, а через f_i – доступную для данного ТСП-соединения пропускную способность в течение передачи i -го сегмента. Тогда

$$f_{i-1} = \frac{\tau r_{i-1}}{\vartheta_{i-1}}. \quad (5)$$

При выборе значения битрейта r_i очередного i -го сегмента полагают, что он должен быть выбран с наибольшим из возможных битрейтов, не превосходящих доступную скорость передачи предыдущего сегмента $r_i = \max(R \leq f_{i-1})$.

Частое изменение значений битрейта в процессе воспроизведения видео может привести к нежелательному ухудшению восприятия передачи. Для уменьшения частоты переключений и исключения статистических выбросов к f_i применяют функцию усреднения $\bar{f}_i = S(f_j : j < i)$.

Для уменьшения возможности прерывания воспроизведения видео на приемной стороне создается буфер, который демпфирует изменения скорости поступления видеосегментов. Здесь следует обратить внимание на то, что процесс управления трафиком осуществляет передающая сторона, получив соответствующую информацию о возникающих потерях пакетов вследствие перегруженности канала.

Управление видеопотоком при технологии HTTP/TSP

Технология, основанная на использовании стека протоколов HTTP/TSP, имеет английское название HAS. Используя весьма надежный протокол TCP, эта технология позволяет осуществлять параллельную передачу видеопрограмм и Internet-данных. Видеопрограммы предварительно преобразуются в цифровую форму и хранятся с различными битрейтами. Это обеспечивает их передачу с различным уровнем качества. Возможность оперативного переключения битрейта используется также для управления интенсивностью видеопотока в зависимости от пропускной способности канала [9].

В настоящее время передача видео составляет почти 90 % от общего объема и HAS становится преобладающей формой передачи Internet-

трафика. Видеопоток HAS также разбивается на отдельные сегменты в несколько секунд времени воспроизведения. Указанные сегменты с различными битрейтами хранятся на сервере и передаются клиентам по их стандартным get-запросам. Здесь следует подчеркнуть, что управление трафиком уже осуществляется не передающей, а приемной стороной. Клиент оценивает пропускную способность ТСП-соединения и запрашивает загрузку следующего сегмента с соответствующим битрейтом.

Вследствие непостоянства пропускной способности канала, предоставляемого пользователю, поступление сегментов происходит неравномерно, в то время как их «потребление» при воспроизведении осуществляется с постоянной битовой скоростью, присущей данному качеству. С целью демпфирования влияния непостоянства пропускной способности на стороне клиента создается некоторый объем пакетов, которые заносятся в буферную память до начала трансляции. Размер буферной памяти измеряется не в числе ячеек памяти, а в единицах времени воспроизведения видео и обычно имеет порядок, равный 10 с. В процессе трансляции наполнение буфера постоянно меняется в зависимости от текущего значения пропускной способности канала ТСП. При длительном уменьшении пропускной способности размер буфера может уменьшиться до нуля и возникнут перерывы в воспроизведении видео. При длительном увеличении пропускной способности число пакетов, накопленное в буфере может превысить размер буферной памяти и часть пакетов будет утеряна, что также приведет к искажению воспроизведения.

Основная задача всех применяемых алгоритмов управления потоками видеотрафика заключается в поддержании размера буфера на заданном уровне и недопущении его опустошения или переполнения. Такие алгоритмы называют алгоритмами адаптации скорости.

Общая модель управления

Рассмотрим один из стандартных алгоритмов управления. Как и прежде, будем считать, что видеопоток разбивается на отдельные сегменты по τ с воспроизведения видео, а каждый сегмент был предварительно подвергнут кодированию в L различных битрейтов видео $R = \{R_1, \dots, R_L\}$. Для каждого клиента процесс потокового вещания поделен на последовательные этапы загрузки $i = 1, 2, \dots$. Этап начинается с момента отправки пользователем i -го запроса и заканчивается началом отправления следующего запроса. Назовем указан-

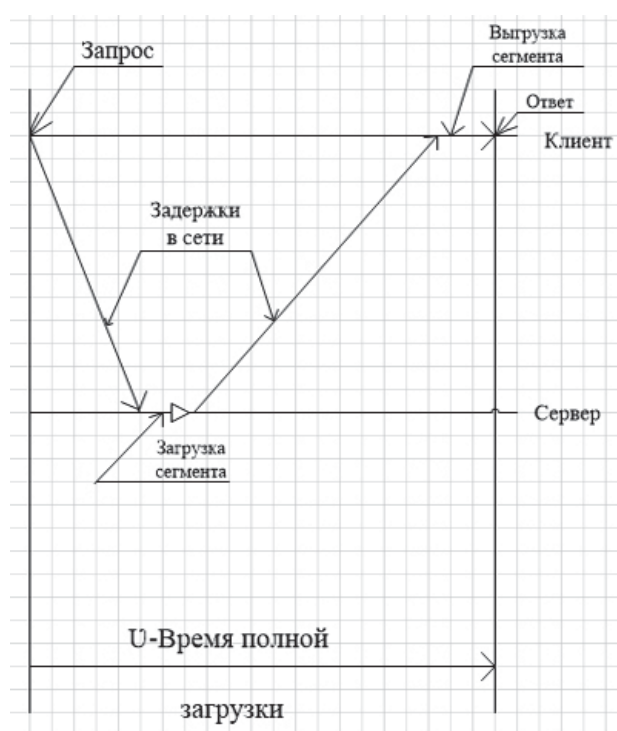


Рисунок 6. Временная диаграмма запросов сегментов

ный интервал времени реальным интервалом T_i между соседними запросами. Обозначим длительность интервала времени между моментом отправления i -го запроса и моментом полной загрузки полученной информации в буфер клиента через ϑ_i , как это показано на рисунке 6.

К началу i -го этапа клиенту известна реальная пропускная способность канала на предыдущем этапе (5). Большинство известных алгоритмов определяет расчетную пропускную способность алгоритма на i -м этапе, приравнивая к ее реальной пропускной способности на предыдущем этапе.

Расчетные значения пропускной способности на соседних этапах могут значительно различаться, поэтому они усредняются на нескольких соседних этапах, что предотвращает влияние отдельных выбросов.

$$\bar{f}_i = S(f_j : j < i), \quad (6)$$

где S – функция усреднения. Усредненное значение пропускной способности квантуется в соответствующее значение битрейта $r_i = Q(\bar{f}_i)$, которое и запрашивается на i -м этапе.

Перед началом каждого этапа загрузки клиент выбирает битрейт очередного сегмента r_i из числа R и определяет расчетное время B_i до начала следующей загрузки: $\theta_i = \frac{\tau r_i}{f_i}$. Если загрузка заканчивается раньше расчетного интервала $\vartheta_i < \theta_i$, то алгоритм ожидает до окончания

интервала $T_i = \theta_i$. В противном случае последующая загрузка начинается сразу же после завершения текущей загрузки $T_i = \vartheta_i$. Именно на основе задания указанной величины происходит управление размером буфера в большинстве алгоритмов.

Обозначим через B_i размер буфера в конце i -го этапа, выраженный в единицах времени воспроизведения видео. В течение всего реального интервала времени i -го этапа T_i размер буфера уменьшается на T_i единиц и увеличивается на τ единиц времени за счет поступившего сегмента. Следовательно,

$$\begin{aligned} B_i &= B_{i-1} + \tau - T_i, \text{ если } B_{i-1} + \tau - T_i \geq 0; \\ B_i &= 0, \text{ если } B_{i-1} + \tau - T_i < 0. \end{aligned} \quad (7)$$

Второе условие соответствует полному опустошению буфера и приводит к перерывам трансляции.

Будем считать, что в результате процесса регулирования буфер никогда не опустошается и всегда выполняется условие первого уравнения. Таким образом, при постоянном времени τ сегмента время T_i до начала следующего запроса по отношению к моменту возникновения предыдущего является единственной величиной, с помощью которой можно управлять размером буфера. Система будет находиться в равновесном состоянии, если выполняется условие $T_i = \tau$.

В некоторых системах управляющий сигнал T_i содержит также информацию о размере буфера B_i на i -м этапе. Так, например, время T_i до начала следующего запроса иногда выбирается на основании соотношения

$$\begin{aligned} T_i &= 0, \text{ если } B_{i-1} < B_{\max}; \\ T_i &= \tau, \text{ если } B_{i-1} \geq B_{\max}, \end{aligned}$$

где $B_{\max}$ – некоторое предельное значение буфера. Имеются алгоритмы (например, PANDA [10]), в которых при формировании T_i учитываются обе указанные составляющие, что увеличивает качество процесса регулирования:

$$T_i = \frac{\tau r_{i-1}}{f_i} + \beta(B_{i-1} - B_{\min}). \quad (8)$$

Здесь β – постоянный коэффициент, учитывающий глубину введенной обратной связи. Первое слагаемое вносит регулирующее воздействие, учитывающее скорость изменения регулируемой величины, а второе учитывает отклонение регулируемой величины (размера буфера B_{i-1}) от некоторого заданного значения $B_{\min}$. Известно, что введение отрицательной обратной связи по скорости изменения управляемой величины улучшает качество процесса регулирования. Поэтому

введение здесь одновременно двух управляющих сигналов является вполне оправданным.

Рассмотрим разность $\Delta v_i = B_{i-1} - B_{i-2}$, которая характеризует скорость изменения размера буфера на $(i-1)$ -м этапе. С целью уменьшения влияния резких изменений скорости можно произвести ее усреднение на ряде предыдущих этапов $\overline{\Delta v}_i = S(\Delta v_j : j < i)$ – по аналогии с формулой (6). Мы считаем возможным значение реального времени T_i до начала следующего запроса определять из условия

$$T_i = \max(\vartheta_i; (\beta \Delta B_i - \alpha \overline{\Delta v}_i)), \quad (9)$$

где α и β – постоянные коэффициенты, определяющие глубину отрицательной обратной связи, обеспечивающие сходимость и устойчивость процесса регулирования, а $\Delta B_i = B_{i-1} - B_{\min}$ – отклонение размеров буфера от заданного минимального значения. Параметр ϑ_i , как и прежде, – длительность интервала времени между моментом отправления i -го запроса и моментом полной загрузки полученной информации в буфер клиента.

Целью рассмотренных выше алгоритмов является поддержание размера буфера B_i в заданных пределах относительно минимального значения $B_{\min}$. Значения B_i относятся к концу i -го этапа. Регулирующий параметр T_i является функцией двух переменных: это рассогласование размера очереди ΔB_i и усредненная скорость ее изменения $\overline{\Delta v}_i$ на i -м этапе.

Все значения параметров, входящих в указанную функцию, зависят только от состояния системы на предыдущих этапах и позволяют определить значения размера буфера на текущем i -м этапе. Достоинство предлагаемого алгоритма состоит в том, что оба регулирующих параметра получаются непосредственно из обработки значений регулируемого параметра на предыдущих этапах. Однако в формуле присутствует также интервал времени ϑ_i , который измеряется на каждом этапе и характеризует задержки, возникающие в сети.

Очереди в сети

Из (9) непосредственно следует, что при заданном состоянии системы время полной загрузки ϑ_i является параметром, определяющим значение регулирующего воздействия T_i . Время полной загрузки в значительной степени обуславливается размерами очередей при прохождении пакетов по сети. Отдельные узлы сети рассматриваются как системы массового обслуживания, а средние размеры очередей пакетов в них определяются по известной формуле Хинчина – Поллачека [11].

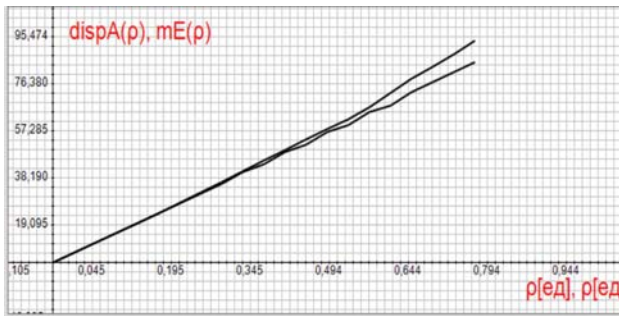


Рисунок 7. Зависимости дисперсии $D_m(\rho)$ – обозначено как $dispA(\rho)$, и числителя обобщенной формулы Хинчина – Поллачека – обозначено как $mE(\rho)$

Однако указанная формула справедлива только для пуассоновских потоков и совершенно непригодна для анализа пачечного видеотрафика. Нами было получено соотношение, обобщающее формулу Хинчина – Поллачека, пригодное для определения средних значений очередей $\overline{q(\rho)}$ для стационарных потоков заявок общего типа [12]:

$$\overline{q(\rho)} = \frac{D_m(\rho) + 2\mu_{q_{i-1}m_i}(\rho)}{2(1-\rho)} - \frac{\rho}{2}, \quad (10)$$

где ρ – коэффициент загрузки; $m_i(\rho)$ – число заявок, поступающих в течение интервала времени обработки одной заявки; $D_m(\rho)$ – дисперсия $m_i(\rho)$, а $\mu_{q_{i-1}m_i}(\rho)$ – второй взаимный центральный корреляционный момент последовательностей $q_{i-1}(\rho)$ и $m_i(\rho)$, называемый ковариацией.

Рассмотрим указанную формулу применительно к предложенной модели группового пуассоновского потока. Дисперсия $D_m(\rho)$ определяется соотношением (4), а $\mu_{q_{i-1}m_i}(\rho) = 0$ ввиду независимости заявок в рассматриваемом потоке. Следовательно, для рассматриваемых групповых пуассоновских потоков получаем простое соотношение

$$\overline{q(\rho)} = \frac{D_m(\rho)}{2(1-\rho)} - \frac{\rho}{2} = \frac{\bar{\rho}k(1+v_k^2)}{2(1-\rho)} - \frac{\rho}{2}. \quad (11)$$

На рисунке 7 показаны зависимости дисперсии $D_m(\rho)$ (обозначено $dispA(\rho)$) и числителя обобщенной формулы Хинчина – Поллачека (обозначено $mE(\rho)$), полученные в результате имитационного моделирования группового пуассоновского потока.

Характеристики практически совпадают. Видно, что дисперсия имеет строго линейную зависимость от коэффициента загрузки ρ . Различия связаны с погрешностями моделирования. Совпадение характеристик подтверждает указанное нами равенство нулю ковариационной составляющей $\mu_{q_{i-1}m_i}(\rho)$.

Мультиплексирование видеопотоков

Мультиплексирование нескольких видеопотоков приводит к разделению между ними имеющегося ресурса и, следовательно, к увеличению суммарного коэффициента загрузки.

Поскольку в качестве модели видеотрафика нами принят групповой пуассоновский поток, рассмотрим мультиплексирование таких потоков. Нетрудно показать, что при суммировании n независимых групповых пуассоновских потоков коэффициент загрузки суммарным потоком равен сумме коэффициентов загрузки отдельных потоков. Суммарный поток, так же как одиночный, является групповым пуассоновским потоком.

При суммировании таких независимых потоков их дисперсии суммируются. Следовательно, в соответствии с (11) средний размер очереди суммы n таких потоков определится соотношением

$$\overline{q(\rho)} = \frac{\sum_{i=1}^n D_m(\rho_i)}{2(1-\rho)} - \frac{\rho}{2} = \frac{E\rho}{2(1-\rho)} - \frac{\rho}{2}, \quad (12)$$

где

$$\rho = \sum_{i=1}^n \rho_i, \quad E_i = \bar{k}_i(1+v_{ki}^2),$$

$$E = \sum_{i=1}^n E_i \frac{\rho_i}{\rho} = \sum_{i=1}^n E_i P_i,$$

где P_i – вероятность загрузки i -м потоком. Следовательно, с точки зрения средних значений очередей суммарный поток эквивалентен групповому потоку с коэффициентом загрузки ρ , равным сумме коэффициентов загрузки исходных потоков, и с коэффициентом E , равным сумме коэффициентов исходных потоков, с учетом вероятностей загрузки этими потоками. При суммировании n одинаковых групповых потоков значения коэффициентов одинаковы: $E = E_i$ и

$$\rho = \sum_{i=1}^n \rho_i = n\rho_i, \text{ следовательно,}$$

$$\overline{q(\rho)} = \frac{En\rho_i}{2(1-n\rho_i)} - \frac{n\rho_i}{2} = \frac{E\rho}{2(1-\rho)} - \frac{\rho}{2},$$

где ρ_i – коэффициент загрузки одним потоком. Мы видим, что при суммировании одинаковых независимых потоков вид характеристики $\overline{q(\rho)}$ остается неизменным.

На рисунке 8 показаны зависимости средних размеров очередей от коэффициента загрузки ρ для группового пуассоновского потока с интенсивностью $\lambda = 100$ заявок в секунду и для суммы трех одинаковых таких потоков. Совпадение очевидно.

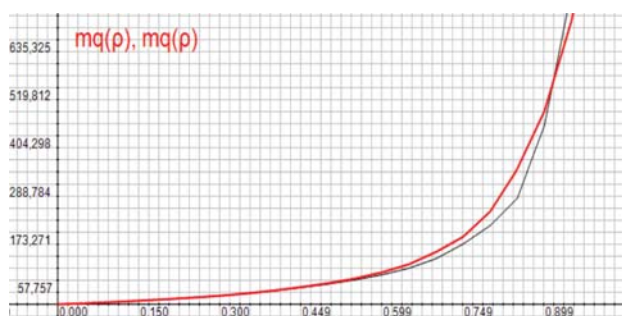


Рисунок 8. Зависимости средних размеров очередей от коэффициента загрузки ρ для группового пуассоновского потока и для суммы трех одинаковых потоков

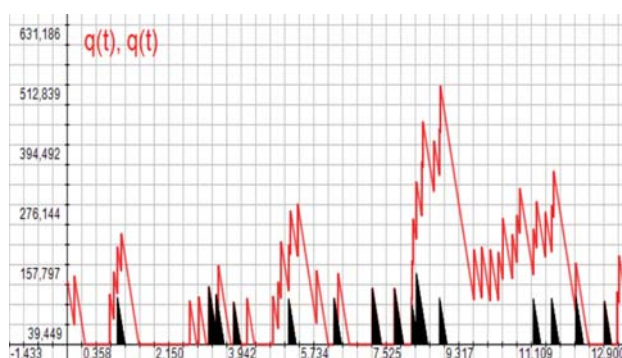


Рисунок 9. Зависимости размеров очередей от времени для группового пуассоновского потока и для суммы трех одинаковых потоков

На рисунке 9 показаны зависимости размеров очередей от времени для одного группового пуассоновского потока $\rho = 0,2$ (более темная заливка) и для суммы трех одинаковых таких потоков ($\rho = 0,6$) – более светлая заливка.

Из рисунка 8 следует, что средний размер очереди $q(\rho = 0,6)$ составляет приблизительно 100 заявок. Максимальное число заявок в очереди суммарного потока, показанное на рисунке 9, при этой же загрузке составляет 513, что почти в 5 раз превышает средний размер очереди.

Фоновый пуассоновский поток

Рассмотрим прохождение видеотрафика совместно с фоновым пуассоновским потоком (13), который представляет частный случай групповых пуассоновских потоков.

На рисунке 10 показаны суммарный трафик кодека H264-10 000 с коэффициентом загрузки $\rho = 0,1$ и пуассоновского потока с интенсивностью $\lambda = 1000$ пакетов в секунду при коэффициенте загрузки $\rho = 0,8$. Потоки имеют одинаковые приоритеты обслуживания. Пачечный трафик кодека H264-10 000 хорошо заметен на фоне равномерного пуассоновского потока.

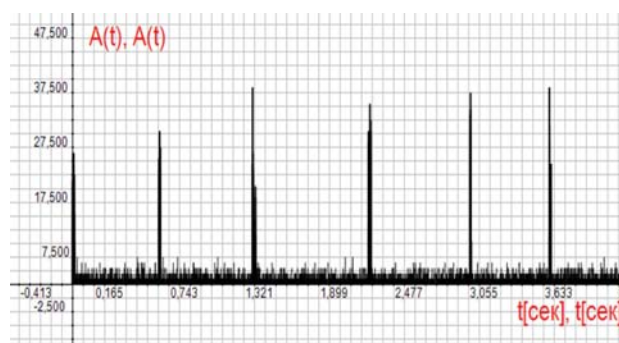


Рисунок 10. Суммарный трафик кодека H264-10 000 и пуассоновского потока

В случае когда имеем обычный пуассоновский поток, число заявок в пачке k постоянно и равно единице, $v_k^2 = 0$, $E = 1$, получаем формулу Хинчина – Поллачека в ее обычном виде:

$$\overline{q(\rho)} = \frac{\rho}{2(1-\rho)} - \frac{\rho}{2} = \frac{\rho^2}{2(1-\rho)}. \quad (13)$$

Поскольку видеопоток и фоновый пуассоновский поток взаимно независимы, дисперсия суммарного потока равна сумме дисперсий исходных потоков. Коэффициент загрузки суммарного потока также равен сумме коэффициентов загрузки исходных потоков. Обозначим через ρ_1 и ρ_2 коэффициенты загрузки видеопотоком и пуассоновским потоком соответственно, $\rho = \rho_1 + \rho_2$ – суммарный коэффициент загрузки. Получим для суммарного потока при беспriorитетном обслуживании:

$$\overline{q(\rho)} = \frac{E_1 \rho_1 + \rho_2}{2(1-\rho)} - \frac{\rho}{2}. \quad (14)$$

Мы видим, что при большой пачечности видеопотока увеличение среднего размера очереди происходит в основном за счет роста суммарного коэффициента загрузки, вызванного наличием фонового пуассоновского потока. При этом числитель увеличивается незначительно.

На рисунке 11 показаны изменения размеров очередей для указанного кодека при отсутствии пуассоновского трафика (черные треугольники) и изменения размеров очередей для указанного трафика, суммированного с пуассоновским потоком (серые треугольники). Внизу едва заметны очереди отдельно взятого фонового пуассоновского потока (черным). Оставшиеся в очереди пакеты пуассоновского потока предыдущего цикла начнут обрабатываться сразу же в начале текущего цикла, смещая начало обработки пачки видеопакетов. После окончания обработки указанной пачки снова начнется обработка пакетов пуассоновского потока.

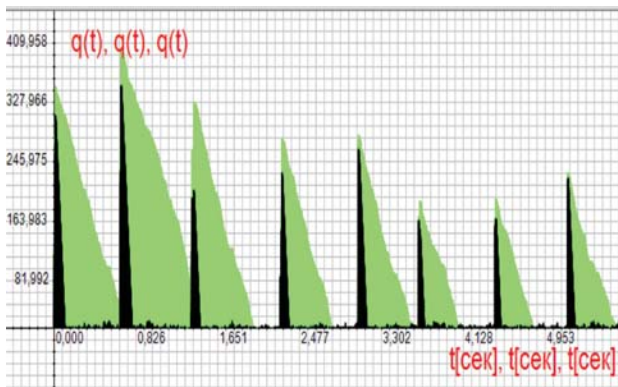


Рисунок 11. Изменения размеров очередей для кодека H264-10 000 при отсутствии пуассоновского потока (черные треугольники) и для суммарного потока (серые треугольники)

Если же пуассоновский поток имеет приоритет в обслуживании по сравнению с видеопотоком, то сначала обрабатываются все пуассоновские пакеты, и лишь после окончания их обработки начнется обработка пакетов видеотрафика. Задержка пачек видеотрафика увеличивается, и фоновый пуассоновский поток, действующий совместно с пачечным потоком, может в несколько раз увеличить время задержки пачек в очередях.

Обозначим через ρ_{UDP} коэффициент загрузки высокоприоритетным потоком UDP, а через ρ_{VID} – коэффициент загрузки пачечным видеопотоком. Тогда эквивалентная загрузка видеопотоком при наличии фонового потока $\rho_{VID/UDP} = \frac{\rho_{TCP}}{1 - \rho_{UDP}}$. В рассмотренном примере $\rho_{TCP} = 0,1$;

$\rho_{UDP} = 0,8$; следовательно, эквивалентная загрузка видеопотоком $\rho_{VID/UDP} = 0,5$.

На рисунке 12 показаны для сравнения очереди суммарного потока (серые треугольники) и трафика рассматриваемого видеокодека (сплошные линии) при эквивалентном коэффициенте его загрузки, равном 0,5. Наблюдается вполне удовлетворительное их совпадение.

Задержки в очередях

Средние размеры очередей определяют и средние значения задержек. Из обобщенной формулы мы видим, что размеры очередей зависят от коэффициента загрузки ρ , который должен учитывать степень загруженности сети другими потоками трафика, в особенности пакетами UDP, имеющими более высокий приоритет. Среднее время задержки заявок в очередях $\bar{w}(\rho) = \bar{q}(\rho)\tau$ имеет порядок 0,2 с. Так обычно и принято считать средние задержки при мультиплексировании видеопотоков. Однако видеотрафик представляет поток отдельных пачек, образующих сегменты.

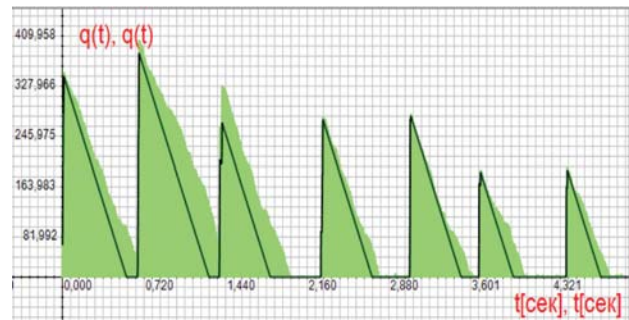


Рисунок 12. Очереди суммарного потока и трафика видеокодека при $\rho_{VID/UDP} = 0,5$

Задержка сегмента в очереди начинается с момента появления пачки, образующей этот сегмент, и заканчивается выходом из очереди последнего пакета данного сегмента (с учетом обработки всех ранее поступивших пакетов).

Указанные задержки определяются максимальными, а не средними значениями возникающих очередей. Максимальная задержка может превышать 1 с. Поэтому задержки пачек пакетов видеотрафика могут быть соизмеримы с временем воспроизведения одного сегмента видеотрансляции, и пренебрегать этими задержками при управлении видеопотоком не следует.

В рассмотренных нами алгоритмах управления величина задержки ϑ непосредственно влияет на сигнал в цепи обратной связи. Целесообразно найти такой алгоритм, при котором указанные задержки исключаются из цепи обратной связи контура управления, что должно улучшить качество процесса регулирования.

Предлагаемый алгоритм

Целью рассмотренных выше алгоритмов является поддержание размера буфера B_i в заданных пределах по отношению к номинальному значению. Обозначим номинальное значение размера буфера через B_0 , а $\Delta B_i = B_i - B_0$ есть отклонение размера буфера от номинального значения. Разность $B_{i-1} - B_{i-2} = \Delta v_i$ принимается за скорость изменения размера буфера на i -м этапе.

Известно, что введение отрицательной обратной связи не только по регулируемой величине, но также и по скорости ее изменения улучшает процесс регулирования. Поэтому предлагается в алгоритме использовать обе указанные величины.

Система будет находиться в устойчивом равновесном состоянии, если величины ΔB_i и Δv_i равны нулю. Недостатком рассмотренных выше алгоритмов является то, что случайная величина, представляющая задержку ϑ_i в канале сети, входит в цепь обратной связи системы регулирования. Предлагаемый алгоритм исключает указанный недостаток.

Алгоритм управления таков.

1. Выбирается номинальное значение битрейта r_0 .
2. Выбирается номинальное значение размера буфера B_0 (приблизительно 10 с).
3. Выбирается значение τ (приблизительно 2 с).
4. Производится запрос с битрейтом r_0 .
5. Определяется реальная пропускная способность канала $f_{i-1} = \frac{\tau r_{i-1}}{\theta_{i-1}}$.
6. Производится усреднение $\bar{f}_i = S(f_j : j < i)$.
7. Выбирается значение битрейта $r_i = Q(\bar{f}_i)$.
8. Измеряются ΔB_i и Δv_i .
9. Выбирается время T_i для очередного запроса согласно таблице $T_i = \Phi(\Delta B_i; \Delta v_i)$.
10. Производится очередной запрос с битрейтом r_i .
11. Переход к п. 5.

В отличие от существующих алгоритмов, информация о средней пропускной способности канала применяется в предлагаемом алгоритме только для определения значений битрейта. Информация, используемая для управления, поступает исключительно за счет обработки значений наполненности буфера.

Заключение

Предложенный способ использования группового пуассоновского потока в качестве модели потокового видеотрафика позволяет получить весьма простые соотношения для его анализа и инженерных расчетов систем управления размерами буферной памяти. Алгоритм, основанный на анализе степени заполнения буферной памяти, дает возможность исключить из обратной связи контура управления задержки, возникающие в сети, что улучшает качество процесса регулирования.

Литература

1. History of Move Networks. URL: <http://movenetworks.com/history.html>
2. Ha S., Rhee I., Xu L. CUBIC: A new TCP-friendly high-speed TCP variant // SIGOPS Oper. Syst. Rev. 42. 2008. P. 64–74.
3. Advanced video coding for generic audiovisual services. Recommendation ITU-T H.264. Geneva: International Telecommunication Union, 2016. 807 p.
4. Вишневский В.М., Дудин А.Н. Системы массового обслуживания с коррелированными

Таблица. Значения функции $T_i = \Phi(\Delta B_i; \Delta v_i)$

ΔB_i	Δv_i	T_i
$\Delta B_i < -\tau$	$\Delta v_i < 0$	$0,25\tau$
	$\Delta v_i \geq 0$	$0,5\tau$
$-\tau \leq \Delta B_i \leq 0$	$\Delta v_i < 0$	$0,75\tau$
	$\Delta v_i \geq 0$	τ
$0 < \Delta B_i \leq \tau$	$\Delta v_i < 0$	τ
	$\Delta v_i \geq 0$	$1,5\tau$
$\Delta B_i > \tau$	$\Delta v_i < 0$	$1,75\tau$
	$\Delta v_i \geq 0$	2τ

входными потоками и их применение для моделирования телекоммуникационных сетей // Автоматика и телемеханика. 2017. Вып. 8. С. 3–59.

5. Дудин А.Н., Клименок В.И. Системы массового обслуживания с коррелированными потоками. Минск: БГУ, 2000. 175 с.
6. Modeling the YouTube stack: From packets to quality of experience / F. Wamser [et al.] // Computer Networks. 2016. Vol. 109. № 2. P. 211–224.
7. Appenzeller G., Keslassy I., McKeown N. Sizing router buffers // SIGCOMM'04. Portland, Oregon, USA, Aug. 30 – Sept. 3. 2004. P. 281–292.
8. Варианты пуассоновского потока событий. URL: <https://studfile.net/preview/7316586/page:7>
9. Lee J.-B., Kalva H. The VC-1 and H.264 Video Compression Standards for Broadband Video Services. Berlin: Springer Science + Business Media, 2008. 515 p.
10. Probe and adapt: Rate adaptation for HTTP video streaming at scale / Z. Li [et al.] // IEEE Journal on Selected Areas in Communications. 2014. Vol. 32. № 4. P. 719–733.
11. Kleinrock L. Queueing computer systems. Vol. 2. Hoboken: John Wiley & Sons, 1976. 576 p.
12. Лихтциндер Б.Я. Трафик мультисервисных сетей доступа (интервальный анализ и проектирование). М.: Горячая линия – Телеком, 2018. 290 с.

Получено 15.05.2020

Лихтциндер Борис Яковлевич, д.т.н., профессор кафедры сетей и систем связи Поволжского государственного университета телекоммуникаций и информатики, 443010, Российская Федерация, г. Самара, ул. Л. Толстого, 23. Тел. +7 846 333-47-69. E-mail: lixt@psuti.ru

STREAM VIDEO TRAFFIC CONTROL ALGORITHMS

Likhttsinder B.Ya.

Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation

E-mail: lixt@psuti.ru

Algorithms for controlling stream video traffic are considered. The features of video codec traffic, which has a pronounced packet character, are analyzed. The characteristics of such flows and their influence on the size of queues and delays in telecommunication network nodes are investigated. A generalized Khinchin-Pollachek formula is given and linear dependence of the formula numerator on the load factor is shown. The main causes of packet delays in the queues of a telecommunication network are considered and the influence of these delays on the processes of controlling stream video traffic is shown. A video stream control algorithm is proposed based on size measurements of the buffer memory in the client's equipment. It is shown that such an algorithm eliminates the influence of delays from the feedback loop of the control system. The results of simulation modeling of the considered processes are presented.

Keywords: *video traffic, control, delays, queues, bit rate, algorithm, videocodecs, group streams, modeling*

DOI: 10.18469/ikt.2020.18.3.09

Likhttsinder Boris Yakovlevich, Povolzhskiy State University of Telecommunications and Informatics, 23, L. Tolstoy Street, Samara, 443010, Russian Federation; Professor of Networks and Communication Systems Department, Doctor of Technical Science. Tel. +7 846 333-47-69. E-mail: lixt@psuti.ru

References

1. *History of Move Networks*. URL: <http://movenetworks.com/history.html>.
2. Ha S., Rhee I., Xu L. CUBIC: A new TCP-friendly high-speed TCP variant. *SIGOPS Oper. Syst. Rev.* 42, 2008, pp. 64–74.
3. *Advanced video coding for generic audiovisual services. Recommendation ITU-T H.264*. Geneva: International Telecommunication Union, 2016, 807 p.
4. Vishnevskij V.M., Dudin A.N. Queuing systems with correlated input flows and their application for modeling telecommunication networks. *Avtomatika i telemekhanika*, 2017, vol. 8, pp. 3–59. (In Russian.)
5. Dudin A.N., Klimenok V.I. *Queuing Systems with Correlated Flows*. Minsk: BGU, 2000, 175 p. (In Russian.)
6. Wamser F. et al. Modeling the YouTube stack: From packets to quality of experience. *Computer Networks*, 2016, vol. 109, no. 2, pp. 211–224.
7. Appenzeller G., Keslassy I., McKeown N. Sizing router buffers. *SIGCOMM'04*, Portland, Oregon, USA, Aug. 30 – Sept. 3, 2004, pp. 281–292.
8. *Poisson event flow variants*. URL: <https://studfile.net/preview/7316586/page:7>.
9. Lee J.-B., Kalva H. *The VC-1 and H.264 Video Compression Standards for Broadband Video Services*. Berlin: Springer Science + Business Media, 2008, 515 p.
10. Li Z. et al. Probe and adapt: Rate adaptation for HTTP video streaming at scale. *IEEE Journal on Selected Areas in Communications*, 2014, vol. 32, no. 4, pp. 719–733.
11. Kleinrock L. *Queueing computer systems. Vol. 2*. Hoboken: John Wiley & Sons, 1976, 576 p.
12. Likhttsinder B.Ya. *Traffic of Multiservice Access Networks (Interval Analysis and Design)*. Moscow: Gorjachaja linija – Telekom, 2018, 290 p. (In Russian.)

Received 15.05.2020