

КАЧЕСТВО ОБСЛУЖИВАНИЯ В СИСТЕМАХ МУЛЬТИМЕДИА

А.Г. КОПАЧЕВ

Работа посвящена разработке программной модели обеспечения качества обслуживания в системах мультимедиа. Предложен метод определения приоритетов мультимедийных потоков в виде графа. На основе предложенной модели разработан эффективный алгоритм межпотоковой адаптации, который распределяет доступную полосу пропускания канала связи между потоками, принадлежащими данному приложению мультимедиа.

Непрерывные потоки данных (звук, видео) представляют собой важный компонент распределенных систем мультимедиа. Таким потокам присущи специфические особенности. Для них критичны временные задержки и емкость канала связи [1–4].

Многие разработки в области распределенных систем мультимедиа основаны на технологии ATM, которая обеспечивает гарантированные параметры временных задержек и пропускной способности [5–7]. Однако такая технология является дорогостоящей для использования в корпоративных образовательных сетях. Поэтому наши исследования сфокусированы в области локальных сетей с пропускной способностью 10-100 Мб/с на базе технологий IP/Ethernet и FDDI [8–11].

Основная проблема в таких средах — перегрузки в сети и высокая загруженность конечных систем (рабочих станций)[12,13]. В результате этого снижается качество обслуживания [14] конечного пользователя. На транспортном уровне невозможно определить характер потоков мультимедиа и адаптация качества обслуживания для разных потоков может быть реализована в равных пропорциях в то время, как на самом деле потоки имеют разные приоритеты.

Современные распределенные приложения мультимедиа (РПМ)[12] строятся на базе взаимодействия типа $M : N$. В таких системах порождается большое количество потоков, взаимодействующих между собой. В зависимости от выполняемой функции каждому потоку назначается приоритет. Для сбалансированной работы всей системы необходимо найти компромисс в изменении качества одного потока в пользу другого. Таким образом, можно сделать вывод о том, что в распределенных приложениях мультимедиа необходимо введение единого механизма контроля за характеристиками потоков данных.

Для решения данной задачи предлагается концепция «качества сессии». Качество сессии, воспринимаемое конечным пользователем, формируется на прикладном уровне. При этом, наиболее важным предположением является то, что потоки, принадлежащие мультимедиа приложению, должны рассматриваться как единое целое, в противном случае, если каждый поток обрабатывается отдельно от других, мы получим множество

конкурирующих за системные ресурсы потоков, что приведет к снижению качества всего приложения.

По мнению автора, реализация механизма мониторинга [2] на прикладном уровне — это один из наиболее эффективных методов решения задачи обеспечения требуемого качества обслуживания в приложениях мультимедиа.

1. ОСОБЕННОСТИ РАСПРЕДЕЛЕННЫХ СИСТЕМ МУЛЬТИМЕДИА

Необходимо отметить особенности распределенных систем мультимедиа, на основе которых у конечного пользователя формируется представление о работе всей системы в целом. Некоторые из этих особенностей достаточно очевидны и описаны, например, в публикациях [13, 14]. По мнению автора, единый подход к определению особенностей систем мультимедиа позволит четко сформировать цели и задачи по обеспечению заданного уровня качества обслуживания для конечного пользователя.

1. Динамический характер. В распределенных приложениях мультимедиа потокам присущи исключительно динамические изменения. Поток может быть создан или остановлен в любой момент, кроме того, относительные приоритеты потоков изменяются во времени. Приоритет потока в конкретный момент времени является функцией от множества активных потоков. Например, в момент времени t , у потока s_1 более высокий приоритет, чем у s_2 . При нехватке ресурсов в системе необходимо изменить параметры потоков для сохранения работоспособности. Учитывая приоритеты, в данном случае, необходимо обеспечить качество потока s_1 за счет изменения характеристик потока s_2 .

2. Степень синхронизованности. Ключевым звеном функционирования распределенных систем мультимедиа является единое представление о синхронизованности потоков всеми пользователями (телемедицина, телеобучение) [2]. Общеизвестного термина синхронизованности не существует, так как он может рассматриваться с разных точек зрения. В некоторых публикациях [3, 4, 14,] под этим термином понимается общее восприятие пользователем распределенной мультимедийной системы. Автором данный термин рассматривается на уровне взаимодействия потоков данных (транспортный), при котором наиболее важным критерием является скорость. Таким образом, если все потоки в системе могут работать на скорости самых медленных узлов или частей сети, и при этом, существует количество участников, не ниже заданного количества, то обеспечивается синхронизованность потоков в системе.

3. Перегрузки. Возникновение перегрузок в сети или на конечной станции ведет к увеличению временных задержек и к потере пакетов. Следовательно, приложения мультимедиа должны иметь механизмы восстановления пакетов в реальном режиме времени без ущерба для пользователей. В публикациях [15, 16] предлагается решать данную проблему для каждого потока в отдельности, в то время, как было отмечено выше, целесообразно управлять потоками вместе с целью поддержания заданного качества обслуживания приложения в целом.

С учетом перечисленных выше особенностей цель данной работы — разработка программной модели, которая бы обеспечивала требуемый уровень качества обслуживания на протяжении всей сессии приложения мультимедиа. Данная модель должна функционировать на границе транспортного и прикладного уровней эталонной модели ISO и учитывать динамическое изменение параметров потоков.

1.1 Управление качеством сессии

При работе распределенных приложений мультимедиа каждый конечный узел подключается к опорной сети. Такое подключение называют сессией или сеансом [2].

Функциональная схема модели управления качеством сессии (УКС) представлена на рис. 1. На подуровне контроля формируется (в т.ч. задается приложением) основная информация для дальнейшего принятия решения об адаптации потоков.

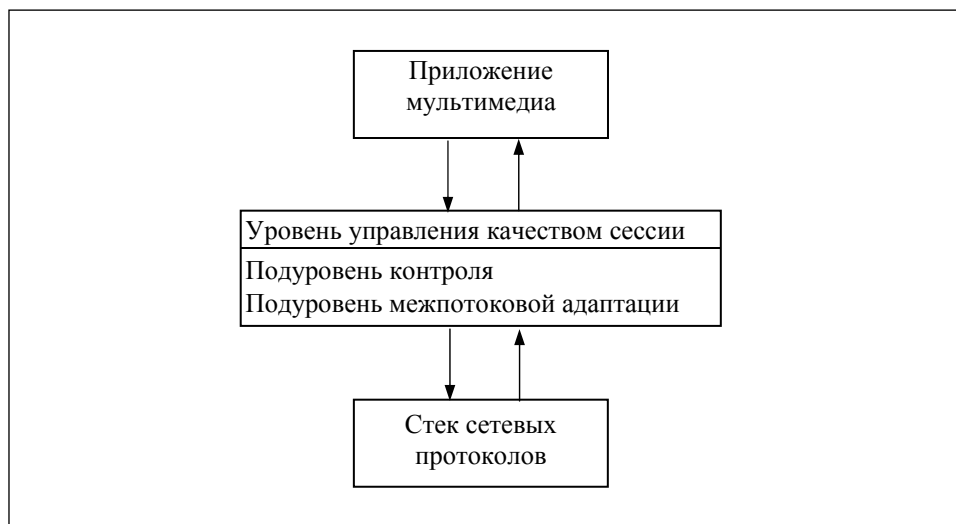


Рис. 1

Общая схема работы УКС представлена на рис. 2. Данная модель функционирует как система обратной связи с замкнутым контуром [6] и постоянно отслеживает состояние потоков, принимает решение относительно адаптации уровня качества обслуживания, устанавливает новые параметры для каждого потока. УКС состоит из двух частей: подуровней контроля и межпоточковой адаптации.

Решение общей задачи обеспечения УКС можно разбить на несколько связанных между собой подзадач [9].

1. Оценить качество потока в каждом узле. Качество каждого потока оценивается с точки зрения его получателя. Это требует наличия контрольного модуля (датчика) на каждом узле.

2. Общая оценка качества потока. Значения характеристик потока могут быть различными у разных узлов в силу их загрузки и межузловых соединений. Поэтому необходимо формирование единых для всех узлов характеристик данного потока. Для решения этой задачи вводится контрольный

модуль, который суммирует представления всех узлов о потоке и принимает решение об изменении характеристик.

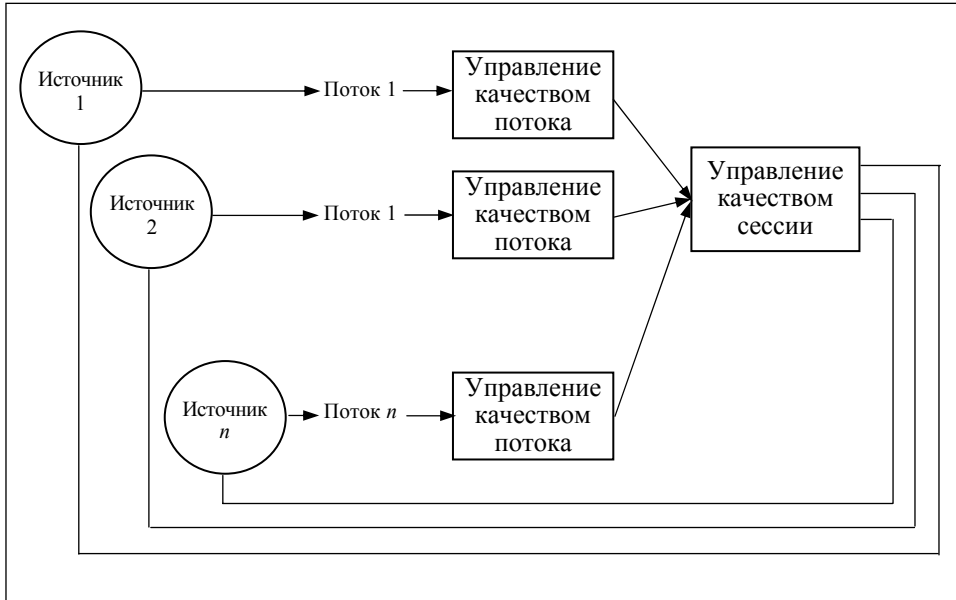


Рис. 2

3. Межпоточковая адаптация. На основе информации о качестве и относительных приоритетах потоков модуль адаптации принимает и формирует рабочие параметры каждого потока таким образом, чтобы достигалось максимальное значение качества сессии всего приложения.

4. Взаимодействие с приложением. Приложение мультимедиа и УКС обмениваются необходимыми параметрами. Например, приложение задает для УКС минимальное значение степени синхронизированности потоков, допустимый рабочий диапазон операций, относительные приоритеты потоков. УКС, в свою очередь, возвращает в приложение текущие рабочие параметры каждого потока.

1.2 Подуровень контроля

Рассмотрим случай одного потока. Как было отмечено выше, в системе с каждым узлом связан датчик — программный агент. Управляющий модуль периодически получает информацию от агентов (датчиков) и формирует общее представление о качестве потоков.

1.2.1 Оценка качества потока. Пусть Q — множество возможных значений качества, которыми характеризуется поток. Определим множество Q как [8]:

$$Q = \{ \text{Неприемлемое, Допустимое, Излишнее} \}. \quad (1)$$

Подуровень контроля определяет для каждого потока k , полученного приемником (узлом) j , качество потока SQ_{kj} . Таким образом, $SQ_{kj} \in Q$. Процесс определения SQ_{kj} будет описан ниже. Суммарная оценка качества

(OSQ_{kj}) потока k формируется на основе множества $\{SQ_{kj}\}$ для $1 \leq j \leq N$, где N — количество приемников, и на основе параметра синхронизации M , представляющего собой минимальное число узлов, в которых есть единое синхронизированное представление о потоке k . Таким образом,

$$OSQ_k = F(SQ_{k1}, SQ_{k2}, \dots, SQ_{kN}, M). \quad (2)$$

Функция F возвращает значение для OSQ_k из множества Q , т.е

$$OSQ_k \in Q. \quad (3)$$

Пусть n_u — часть приемников, для которых $SQ_{kj} = \text{Неприемлемый}$, а n_a , — часть приемников, для которых $SQ_{kj} = \text{Допустимый}$. С учетом этих параметров и на основании вышеизложенного запишем формулу F для вычисления OSQ_k в виде:

$$F = \begin{cases} \text{Неприемлемый}, & \text{если } n_u > 1 - M, \\ \text{Допустимый}, & \text{если } n_u + n_a > 1 - M \text{ и } n_u \leq 1 - M, \\ \text{Излишний}, & \text{в противном случае.} \end{cases} \quad (7)$$

Установка параметра M в 1 означает строго синхронизированные представления о потоке k для всех узлов. Если хотя бы один из узлов получает поток k с качеством, ниже остальных (например, меньшая частота обновления видео, большой разброс временных задержек пакетов, большие потери пакетов), то уровню качества для данного потока присваивается наименьшее значение. Это вызывает процесс ресинхронизации параметров потока для данного узла с остальными, что будет подробно описано ниже. С другой стороны, установка значения параметра M в 0, ведет к состоянию полностью несинхронизированных потоков.

1.2.2 Критерии контроля. Достоверность определения параметров качества заданного потока k зависит от правильности определения значения SQ_{kj} для каждого узла j . Это достигается путем введения критериев контроля. Обычно таковыми являются [18]:

- процент потерянных пакетов;
- время, прошедшее с момента получения последнего пакета;
- среднее время отклика;
- колебания времени отклика;
- эффективная пропускная способность.

Среднее время отклика помогает оценить процент потерь пакетов в случаях, когда все пакеты, посланные источником со времени прибытия последнего пакета, потеряны. Данный параметр может быть использован как критерий для оценки времени прохождения пакетов от источника к приемнику и наоборот, что очень важно в случаях несимметричных каналов. Параметр колебания времени отклика J — девиация величины D , представляющая собой относительное время прохождения двух пакетов от источника к приемнику. Если S_i — временная метка, полученная пакетом i

в источнике, и R_i — время прохождения пакета i , то для двух пакетов i и j формулу определения D можно записать в тривиальном виде:

$$D(i, j) = (R_j - R_i) - (S_j - S_i) = (R_j - S_j) - (R_i - S_i). \quad (5)$$

Значение колебания времени отклика для каждого полученного пакета i и значения D для него и предыдущего пакета $i-1$ вычисляются в соответствии с формулой

$$J = J + (|D(i-1, i)| - J) / 16. \quad (6)$$

Согласно [9], коэффициент $1/16$ является оптимальным и подобран экспериментальным путем. Таким образом, параметр J помогает определить перегрузки в сети и избежать потери пакетов.

Как правило, каждый из критериев контроля применяется с определенным весовым коэффициентом [16]. Например, нам необходимо вычислить значение SQ_{kj} только с использованием коэффициента потери пакетов L , взвешенная величина которого определяется так: $L = \alpha l + (1 - \alpha)L$, где $0 \leq \alpha \leq 1$ [18], l — текущее значение коэффициента потери пакетов. Обозначим значения нижней и верхней границ соответственно L_{\min} и L_{\max} . Тогда SQ_{kj} определяется следующим образом:

$$S_{kj} = \begin{cases} \text{Излишний,} & \text{если } L \leq L_{\min}. \\ \text{Допустимый,} & \text{если } L_{\min} < L \leq L_{\max}. \\ \text{Неприемлемый,} & \text{если } L > L_{\max}. \end{cases} \quad (7)$$

1.3 Подуровень межпоточковой адаптации

При изменении ключевых характеристик (адаптации) потоков в качестве основного критерия выберем скорость передачи данных. Таким образом, необходимо решить задачу распределения всей полосы пропускания R_{tot} , доступной приложению, между всеми потоками данного приложения. Такое требование справедливо в условиях неравномерности мультимедийного трафика. С одной стороны, существующая сетевая инфраструктура и большинство операционных систем не обеспечивают гарантированное и эффективное распределение ресурсов. С другой стороны, решение данной задачи осложняется динамическим изменением характеристик потоков.

Таким образом, решить данную проблему можно путем введения процедуры контроля и управления характеристиками потоков. В предлагаемой модели эту функцию выполняет подуровень межпоточковой адаптации, реализованный на прикладном уровне эталонной модели ISO/OSI.

1.3.1. Критерии адаптации. Подуровень адаптации отвечает за выбор ключевых параметров потока и принятие решений по их изменению на основе следующих критериев.

1. Допустимый рабочий диапазон. Для каждого потока k задается допустимый диапазон операций. Такой операцией может быть, например,

скорость R_k , на которой источник пересылает данные. Условие для данного критерия запишем в виде: $R_{\min,k} \leq R_k \leq R_{\max,k}$. Это достаточное условие — источник может передавать данные в диапазоне $[R_{\min,k}, R_{\max,k}]$, но не необходимое, так как может быть задан и дискретный набор значений. Однако очевидно, что в любой момент времени должно выполняться условие $\sum_{i=1}^n R_i \leq R$, где n — количество потоков.

2. Текущие значения параметров — последнее выбранное значение скорости R_k .

3. Постепенное изменение. Для того, чтобы избежать резкого изменения параметров качества потоков, вводим два дополнительных критерия для каждого потока — аддитивный коэффициент улучшения e_k , и мультипликативный коэффициент ухудшения d_k . Таким образом, изменение параметра R_k может быть в пределах $\text{Min}(R_k + e_k, R_{\max,k}) < R_k < \text{Max}(R_k * d_k, R_{\min,k})$, где $e_k > 0$ и $0 < d_k < 1$. В случае возникновения перегрузки в канале, необходимо уменьшить занимаемую полосу пропускания. Для этого используется коэффициент d_k , с другой стороны, для постепенного увеличения емкости полосы пропускания используется e_k . Такой подход основан на процедуре медленного старта, который реализован в протоколе TCP [16].

4. Относительные приоритеты потоков. В приложении задаются относительные приоритеты потоков друг к другу. Таким образом, заданный набор из четырех параметров полностью определяет входные данные для подуровня адаптации потоков.

1.4 Модель качества сессии

Относительные приоритеты мультимедиа потоков представим в виде направленного графа $G(V, E)$, где V — множество вершин, а E — множество ребер. Каждая вершина представляет собой активный поток. Ребро, из вершины i в вершину j означает, что поток i имеет больший приоритет над j в данный момент времени. Следовательно, при возникновении перегрузок часть ресурсов у потока будет отдано потоку i . Для каждой вершины (потока) k обозначим B — множество вершин («заемщики»), у которых k может занимать ресурсы. Необходимо отметить, что если поток i может занимать ресурсы у j , а j у k , то i может занимать их у k , т.е. отношения займа транзитивны. На основании вышеизложенного множество B_k может быть представлено в виде следующей рекурсивной формулы:

$$B_k = \bigcup_{\forall j:(k,j) \in E} (\{j\} \cup B_j). \quad (8)$$

Допустим, что в приложении есть только два типа потоков — звук и видео. Типичный для такого случая граф потоков и их приоритетов представлен на рис. 3. Граф в терминах предложенной модели УКС показан

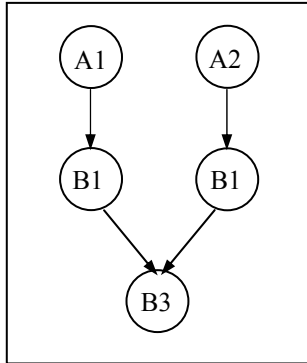


Рис. 3

на рис. 4, где узел «заемщик»; $B3$ — панорамное видео аудитории; A и B — соответственно означают аудио- и видео-потоки.

На рис.5 представлен линейный список потоков в порядке убывания их приоритетов (см. рис. 3).

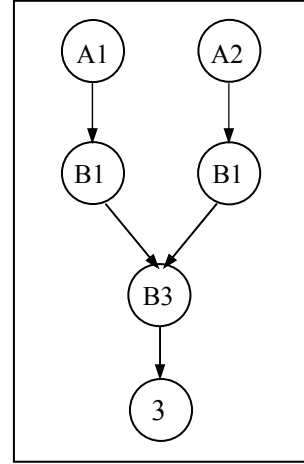


Рис. 4

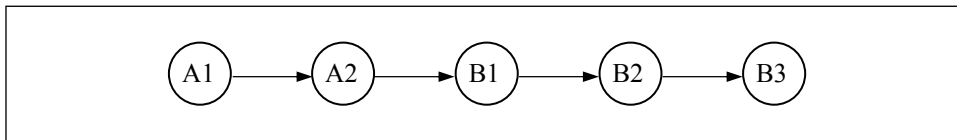


Рис. 5

1.5 Алгоритм межпоточковой адаптации

Ниже описан алгоритм межпоточковой адаптации (АМПА) на основе критерия скорости передачи потоков. Цель данного алгоритма — распределить доступную полосу пропускания канала между потоками таким образом, чтобы достигалось максимальное значение качества сессии.

Предлагаемый алгоритм должен выполняться периодически и работает следующим образом. Если в процессе мониторинга получены данные, что качество потока k $OSQ_k = \text{Неприемлемое}$, то выполняется процедура $\text{Заем}()$, в противном случае — процедура $\text{Увеличение}()$.

Процедура $\text{Заем}()$ выбирает один приоритетный поток, например k , и пытается занять часть полосы у потоков с низким приоритетом из множества B_k . Данная процедура выполняется в два этапа. На первом этапе часть требуемой полосы пропускания занимает у низкоприоритетных потоков с уровнем качества Неприемлемое ; на втором — за счет заема у низкоприоритетных потоков с уровнем качества Излишнее . Этот процесс повторяется для каждого потока с качеством Неприемлемое по убыванию приоритета, начиная с более высокого. За один цикл у потока может быть заимствована часть полосы пропускания только один раз.

Процедура $\text{Увеличение}()$ выбирает приоритетный поток с уровнем качества Излишнее и пытается увеличить ширину его полосы пропускания за счет заема. Этот процесс повторяется для каждого потока со значением

уровня качества *Излишнее* в порядке убывания приоритета до тех пор, пока есть свободная полоса пропускания для заема.

АЛГОРИТМ МЕЖПОТОКОВОЙ АДАПТАЦИИ (АМПА)

Алгоритм АМПА

Вход: $R_{\min,i}, R_{\max,i}, R_i, OSQ_i, e_i, d_i$ для $1 \leq i \leq n$, граф G

Выход: R_i для $1 \leq i \leq n$

Начало

Сохранить Начальные_Значения // $\{R_i\}$

Если существует хотя бы один поток k такой, что $OSQ_k =$
= Неприемлемое

Если Предыдущие_Значения стабильны // Нет потоков с $OSQ_k =$
= Неприемлемое

Вернуться к Предыдущим_Значениям

Иначе

Выполнить процедуру *Заем()*

Иначе

Выполнить процедуру *Увеличение()*

Пусть Предыдущие_Значения=Начальные_Значения

Конец.

Процедура *Заем()*

Начало

Массив_Займа=0

Для каждого потока i с $OSQ_k =$ Неприемлемое, начиная с
самого приоритетного

$Потребность = e_i$

Сформировать множество «заемщиков» B_i

Сформировать множество $U_i \subset B_i$, где $OSQ_j =$ Неприемлемое,
для всех $j \in U_i$

$Заем_u = \sum_{j \in U_i} (R_j - \text{Max}(R_j * d_j, R_{\min,j})) + \text{Массив_Займа}$

$R_j = R_j - \text{Max}(R_j * d_j, R_{\min,j})$ для всех $j \in U_i$

Отметить поток j как обработанный для всех $j \in U_i$

Если $Заем_u \geq Потребности$

$R_i = R_i + Потребности$

Добавить ($Заем_u - Потребности$) в Массив_Займа

продолжить // перейти на следующий цикл итерации

Если $Заем_u < Потребности$

$Потребности = Потребности - Заем_u$

Сформировать множество $O_i \subset B_i$, где $OSQ_j = \text{Излишнее}$,
для всех $j \in O_i$

$$\text{Заем}_o = \sum_{j \in O_i} (R_j - \text{Max}(R_j * d_j, R_{\min, j}))$$

$$R_j = R_j - \text{Max}(R_j * d_j, R_{\min, j}) \text{ для всех } j \in O_i$$

Отметить поток j как обработанный для всех $j \in O_i$

Если $\text{Заем}_o \geq \text{Потребности}$

$$R_i = R_i + \text{Потребности} + \text{Заем}_u$$

Добавить $(\text{Заем}_o - \text{Потребности})$ в Массив_Заема

продолжить // перейти на следующий цикл итерации

Если $\text{Заем}_u < \text{Потребности}$, уменьшить качество потока i

Добавить $(\text{Заем}_u + \text{Заем}_o)$ в Массив_Заема

$$R_i = \text{Max}(R_i * d_i, R_{\min, i})$$

Добавить $R_i - \text{Max}(R_i * d_i, R_{\min, i})$ в Массив_Заема

Конец

Процедура Увеличение()

Начало

$$\text{Массив_Заема} = R_{\text{tot}} - \sum_{i=1}^n R_i$$

для каждого потока i с $OSQ_k = \text{Излишнее}$, начиная с самого приоритетного

$$\text{Увеличение} = \text{Min}(e_i, \text{Массив_Заема}, R_{\max, j} - R_i)$$

$$\text{Массив_Заема} = \text{Массив_Заема} - \text{Увеличение}$$

$$R_i = R_i + \text{Увеличение}$$

Конец.

ВЫВОДЫ

1. Основная идея предложенной модели состоит в том, что все потоки, принадлежащие данному приложению мультимедиа, должны обрабатываться как одно целое. Для этого вводится понятие синхронизированного представления потоков с точки зрения получателей. Таким образом, качество сессии достигается путем отслеживания индивидуальных параметров потоков и их адаптации с целью обеспечения максимального значения их качества в целом. Таким образом, предложенный подход и модель на его основе позволяют расширить функциональность стека протоколов.

2. Уточнена структура модели организации сессии в приложении мультимедиа; для чего введены подуровень контроля и межпоточковой адаптации, описаны их функции. Предложен метод определения приоритетов мультимедийных потоков в виде направленного графа, что значительно упрощает программную реализацию механизма межпоточковой адаптации.

3. Предложен и реализован эффективный алгоритм межпоточковой адаптации, который позволяет распределять доступную полосу пропускания канала связи между потоками, принадлежащими данному приложению мультимедиа.

ЛИТЕРАТУРА

1. Бесслер Р., Дойч А. Проектирование сетей связи. — М.: Радио и связь, 1988. — 277 с.
2. Youssef A., Abdel-Wahab H., Maly K. Inter stream adaptation for collaborative multimedia applications // Proc. of ACM conference. — 1996.
3. Abdel-Wahab H., Maly K., Youssef A. al etc. The software architecture and interprocess communications of IRI // WETICE'96 IEEE 5th workshop on enabling technology. — Stanford. — 1996.
4. Abdel-Wahab H., Feit M.A. XTV: a framework for sharing X window clients in remote synchronous collaboration // Proc. IEEE conf. on communic. software. — 1991. — P. 159–167.
5. Armbruster H. ATM flexibility: future multimedia support and mobil communications // IEEE communications magazine. 1995. — 33, N 2.
6. Tobagi F. High performance network design: looking on to the future // HPN'95. Palma de Mallorca. — 1995.
7. Miguel T., Pavyn S., Quemada J. ISABEL-Experimental Distributed Cooperative Work Application over Broadband Networks // Lecture Notes in Computer Science. — 1994. 868. — P. 353–362.
8. Кульгин М. Технология корпоративных сетей. Энциклопедия. — СПб.: Питер, 1999. — 704 с.
9. Лисмонад Н.И. Моделирование и оптимизация глобальных сетей. — Мн.: БГУ, 2000. — 156 с.
10. Лисмонад Н.И. Синтез оптимальных сетей// Докл. НАН Беларуси. — 2000. — 44, № 2. — С. 37–40.
11. Maly K., Wild C., Overstreet C.M. al etc. Interactive remote instruction // ACM SIGCSE/SIGCUE. — Barcelona, 1996.
12. European Commission. ACTS97 — Advanced Communications and Services // Project Summaries (Ref. No. AC971392-PS). — P.190.
13. Banerjea A., Ferrari D. The Tenet real-time protocol suite: design, implementations and experience //Technical report TR-94-059, International Computer science institute. — Berkeley, USA, 1994.
14. Campbell A., Coulson G., Hutchinson D. A quality of service architecture //Internal report MPG-94-08. Lancaster University. — 1994.
15. Busse I., Deffner B., Schulzrinne H. Dynamic QoS control of multimedia applications based on RTP. // Second workshop on protocols for multimedia systems. — Salzburg, 1995.
16. Talley T., Jeffay K. Two-dimensional scaling techniques for adaptive transmission control of live audio and video streams // Proc. of ACM multimedia. San Francisco. USA. 1994. P. — 247–254.
17. Вязникова М.В. Реализация услуг мультимедиа на широкополосных сетях связи. — М. ЦНТИ «Информсвязь» — 1997.

Поступила 20.01.2002