STUDIA INFORMATICA 2001
Volume 22 Number 2 (44)

Robert R. CHODOREK
Akademia Górniczo-Hutnicza, Katedra Telekomunikacji

STEROWANIE WIELOPUNKTOWĄ TRANSMISJĄ MULTIMEDIALNĄ[1]

Streszczenie. W artykule przedstawiono metody prowadzenia transmisji multimedialnych w sieciach Internet. Szczególną uwagę zwrócono na sterowanie wielopunktową transmisją multimedialną. Przedstawiono również problemy efektywnego zarządzania transmisją multimedialną.

CONTROL OF MULTICAST MULTIMEDIA TRANSMISSION

Summary. In the paper methods of multimedia transmission in Internet was presented. To perform multicast transmission of multimedia data we have to use specialized protocols for control and management of session. In the paper an analysis of efficient management of multicast multimedia transmission was presented.

1. Wprowadzenie

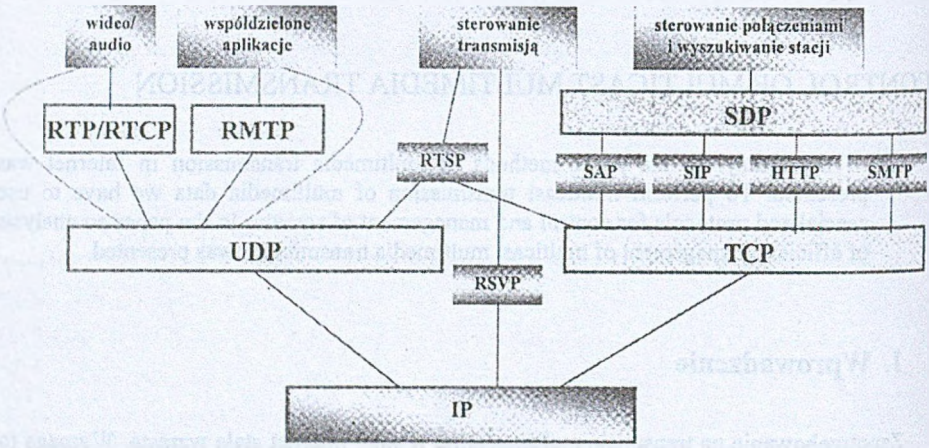
Zapotrzebowanie na transmisję multimedialną w sieci Internet stale wzrasta. Wymaga to dużego pasma transmisyjnego oraz realizacji przez sieć szeregu dodatkowych usług. Istotnym mechanizmem usprawniającym transmisję multimedialną jest transmisja wielopunktowa (ang. *multicast*). Realizacja takiej transmisji wymaga jednak szeregu dodatkowych środków do sterowania i zarządzania transmisją wielopunktową.

W artykule zostanie zaprezentowana współpraca aplikacji multimedialnych z sieciami teleinformatycznymi. Przedstawiona zostanie analiza metod sterowania transmisją wielopunktową. Zaprezentowane zostaną również problemy, jakie występują przy wdrażaniu multimedialnej transmisji wielopunktowej.

[1] Praca została zrealizowana w ramach grantu KBN nr 7 T11D 012 20

2. Transmisja multimediów w sieci Internet

Aplikacje multimedialne łączą wiele usług, a w szczególności transmisję tekstu, dźwięku, obrazów nieruchomych (grafiki), obrazów ruchomych (wideo). Wymagają przy tym dodatkowych, znacznie bardziej skomplikowanych niż miało to miejsce w przypadku tradycyjnej transmisji danych, mechanizmów, np. wymagają one ścisłej synchronizacji oraz nieprzerwanej (strumieniowej) transmisji. Różne są również, wymagane przez poszczególne aplikacje, przepustowości sieci: od małych dla aplikacji tekstowych do bardzo dużych dla transmisji telewizyjnej HDTV. Generalnie wymagają one efektywnej transmisji kilku strumieni danych do wielu odbiorców przy zapewnieniu określonej jakości świadczonych usług transportowych (QoS) [2]. Istotne jest również, aby transmisja odbywała się bez zmiany wielkości opóźnienia pomiędzy kolejnymi pakietami (lub grupami pakietów) przenoszących jeden element przekazu, np. pojedynczą ramkę obrazu.



Rys. 1. Protokoły wykorzystywane do transmisji multimedialnej
Fig. 1. Protocols for multimedia transmission

Realizacja powyższych zadań wymaga użycia wielu protokołów sieciowych. Stos protokołowy wykorzystywany do transmisji multimedialnych w Internecie przedstawiono na rys. 1.

W wielu przypadkach aplikacje samodzielnie oferują dodatkowe usługi związane z transmisją w sieci. Przykładowo, w standardzie MPEG przewidziano procedury związane z pakietyzacją strumienia oraz z synchronizacją strumieni podczas transmisji poprzez sieć. Umożliwiło to zdefiniowanie współpracy standardu MPEG bezpośrednio z siecią ATM. Istnieje jednak duża grupa aplikacji wymagająca bogatego zestawu usług sieciowych.

Oprócz transmisji danych konieczne jest zapewnienie dodatkowych usług. Jednym z ważniejszych zagadnień jest synchronizacja wielu strumieni. Innym jest możliwość sterowania przebiegiem prezentacji multimedialnej.

Do transmisji danych wykorzystywane są protokoły transportowe. W przypadku aplikacji multimedialnych protokół powinien realizować transmisję wielopunktową oraz posiadać możliwość definiowania jakości usług (QoS). Klasyczne protokoły (np. TCP) nie posiadają mechanizmów charakterystycznych dla nowszych rozwiązań, takich jak szybkie zestawianie połączeń czy transmisja wielopunktowa [1]. Różnorodność transmisji multimedialnych powoduje, że stosowanych bywa kilka protokołów transportowych, dostosowanych do konkretnych rozwiązań. Do najpopularniejszych możemy zaliczyć: TCP, UDP oraz RTP. Do sterowania prezentacją multimedialną często wykorzystywany jest protokół RTSP.

Protokół TCP (*Transport Control Protocol*) [8] zaprojektowano dla usług, które można było zrealizować na początku lat 80.: poczta elektroniczna (bez możliwości przekazów multimedialnych), transmisja plików za pomocą FTP czy interaktywna praca na zdalnym komputerze (usługa telnet). Wielu usług nie sposób było wówczas zrealizować na typowym systemie końcowego użytkownika. Również sama sieć stanowiła poważne ograniczenie. Te realia wpłynęły na ostateczny kształt protokołu TCP. Obecnie protokół TCP spełnia rolę protokołu transportowego dla przesyłania sygnałów sterujących przebiegiem sesji multimedialnej. Wykorzystywany jest on m.in. przez protokoły HTTP i RTSP.

Pierwotna specyfikacja protokołu TCP była kilkakrotnie zmieniana. Celem tych zmian było wprowadzenie dodatkowych usprawnień, jak również poprawienie błędów pierwotnej specyfikacji protokołu [6][12]. W efekcie pojawiło się kilka wersji rozwojowych protokołu TCP, z których najbardziej znane są: TCP Tahoe, TCP Reno, TCP New-Reno, TCP Vegas i TCP SACK. Każda z wersji protokołu TCP posiada inne algorytmy powolnego startu, zapobiegania zatorom i inne strategie korekcji błędów. Ma to wpływ na poprawę wydajności protokołu.

Niezależnie jednak od wersji protokół TCP zapewnia niezawodne przesyłanie danych, korzystając z mechanizmu ACK - pozytywnego potwierdzenia. Protokół TCP został zaprojektowany jako protokół transportowy ogólnego przeznaczenia, nie jest wobec tego zoptymalizowany pod kątem przenoszenia konkretnego typu danych. Obecnie jest on wykorzystywany głównie w celu niezawodnej transmisji danych. TCP jest jednym z najczęściej stosowanych protokołów transportowych. Jest on podstawowym protokołem transportowym w sieci Internet.

Najprostszym protokołem wykorzystywanym do transmisji multimedialnych jest protokół UDP (*User Data Protocol*). Jest on protokołem bezpołączeniowym, który realizuje transmisję bez mechanizmów zapewnienia niezawodności transmisji danych [9]. Pozwala na identyfikację aplikacji w systemach docelowych. Protokół może realizować detekcję błędów

za pomocą sum kontrolnych, nie ma jednakże zdefiniowanych mechanizmów korekcji błędów. Nie realizuje również kontroli poprawności otrzymywanego strumienia danych pod względem kolejności otrzymywanych danych (błędna kolejność otrzymywanych danych może wystąpić np. przy innym wyborze trasy dla każdego pakietu). Podobnie jak protokół TCP protokół UDP pracuje na bazie protokołu sieciowego IP [7].

Możliwość zastosowania protokołu UDP do transmisji multimedialnych wynika z braku opóźnień spowodowanych przez mechanizmy retransmisji. Protokół UDP stosujemy również do realizacji aplikacji typu klient-serwer, w których występują krótkie i pojawiające się często komunikaty typu: zapytanie-odpowiedź. Protokół UDP świadczy również usługi dla innych protokołów transportowych jak chociażby RTP.

Protokół RTP (*Realtime Transport Protocol*) [10] jest protokołem transportowym zaprojektowanym do transmisji danych w czasie rzeczywistym. Własności protokołu pozwalają na wykorzystanie go jako protokołu do transmisji danych multimedialnych, takich jak interaktywne usługi audio lub wideo. Najczęściej protokół RTP korzysta z usług protokołu UDP, który zapewnia multipleksację oraz kontrolę błędów. Możliwa jest także współpraca tego protokołu z innymi realizacjami niższych warstw sieci. Protokół RTP przewidziano zarówno dla transmisji typu punkt-punkt (ang. *unicast*), jak również do transmisji wielopunktowych (ang. *multicast*).

RTP wewnętrznie nie zawiera mechanizmów zapewniających właściwe dostarczanie danych (z odpowiednimi parametrami czasowymi). Zapewnia jednak mechanizmy sekwencyjnego dostarczania danych oraz synchronizacji danych. Protokół RTP identyfikuje również rodzaj przesyłanych danych (audio, wideo) oraz metodę kodowania danych (np.: PCM, ADPCM).

Każdy pakiet protokołu RTP posiada znacznik czasowy. Pozwala on na realizację synchronizacji oraz określenie zmienności czasu dostarczenia pakietu - zmienności opóźnień (ang. *jitter*). Wartość startowa znacznika czasowego jest losowa. Przy realizacji prawidłowej transmisji powinien on liniowo wzrastać.

Z protokołem RTP współpracuje protokół RTCP (*Real-Time Control Protocol*) [10]. Realizuje on wiele funkcji pomocniczych niezbędnych do prawidłowej pracy protokołu RTP. Protokół RTCP monitoruje transmisję, określając jej jakość. Informacja ta może być wykorzystana przez aplikację. Oprócz funkcji monitorowania połączeń protokół RTCP identyfikuje uczestników konwersacji. Zapewnia także tworzenie grup konferencyjnych z pełną identyfikacją uczestników. Stan konferencji jest przekazywany na bieżąco. W tym celu protokół RTCP wysyła regularnie pakiety do wszystkich uczestników konferencji, korzystając z tych samych mechanizmów co protokół bazowy RTP.

RTCP spełnia cztery funkcje:

- monitorowanie jakości (QoS) transmisji danych. Na podstawie tej informacji można zrealizować kodowanie adaptacyjne (dostosowywanie jakości kodowanego przekazu do jakości usług świadczonych przez sieć),
- identyfikacja źródła informacji,
- skalowanie sesji. Dzięki tej funkcji każdy użytkownik posiada informacje o liczbie stacji biorących udział w sesji. Liczba ta jest używana do obliczenia prędkości wysyłania pakietów,
- przenoszenie dodatkowej informacji kontrolnej sesji, np. identyfikacja uczestnika w celu zobrazowania w interfejsie użytkownika. Jest to funkcja opcjonalna.

3. Zarządzanie multimedialną sesją wielopunktową

Zarządzanie sesjami wielopunktowymi wymaga mechanizmów do wyszukiwania stacji i sterowania połączeniami. Mechanizmy te powinny umożliwiać stworzenie nowej lub dołączanie się do już istniejącej sesji wielopunktowej oraz na definicję podstawowych parametrów transmisji. W chwili obecnej do tego celu wykorzystywane są m. in. protokoły SDP (*Session Description Protocol*) [4] i SIP (*Session Initiation Protocol*) [5] (rys. 1).

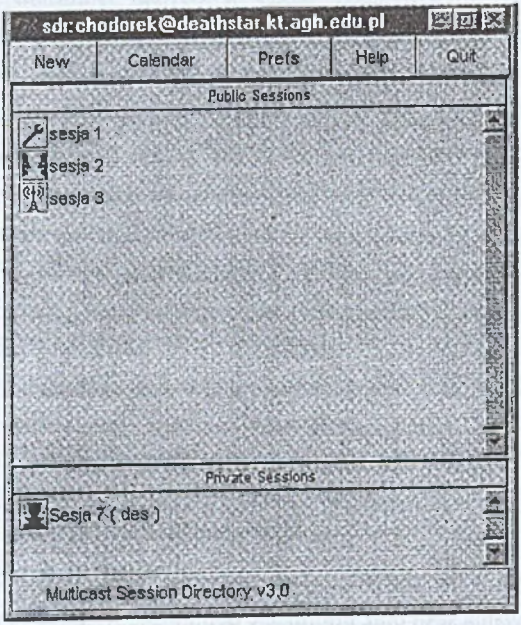
Protokół SDP służy do ogłaszania sesji wielopunktowych i przekazywania parametrów startowych konferencji do wszystkich uczestników. SDP definiuje uniwersalny format opisu sesji. Opis ten przekazuje do uczestników sesji korzystając z jednego z protokołów (rys. 1). W przypadku sesji wielopunktowych dane o sesji rozgłaszane są cyklicznie na predefiniowany wielopunktowy adres IP i numer portu. Możliwe jest również rozgłaszanie sesji za pomocą poczty elektronicznej czy też WWW, korzystając z predefiniowanego rozszerzenia MIME dla SDP.

Dane przekazywane za pomocą SDP obejmują informacje identyfikujące sesję, informacje o czasie trwania sesji oraz informacje definiujące wykorzystywane w sesji media i metody transmisji tych mediów. Oprócz informacji podstawowych można przekazywać szereg parametrów opcjonalnych. Pozwalają one na zdefiniowanie prędkości bitowych strumienia multimedialnego, jakości transmisji oraz szeregu pomocniczych ustawień jak chociażby definicję języka, w którym odbywa się sesja.

Protokół SIP służy do nawiązania, zmodyfikowania i zakończenia sesji multimedialnej. Wykorzystywany jest do zarówno do sesji wielopunktowych, jak i punkt-punkt. Realizuje operacje związane z lokalizacją uczestników sesji, określeniem mediów transmisyjnych, jak i parametrów transmisji. Informacje o sesji najczęściej przekazuje używając protokołu SDP.

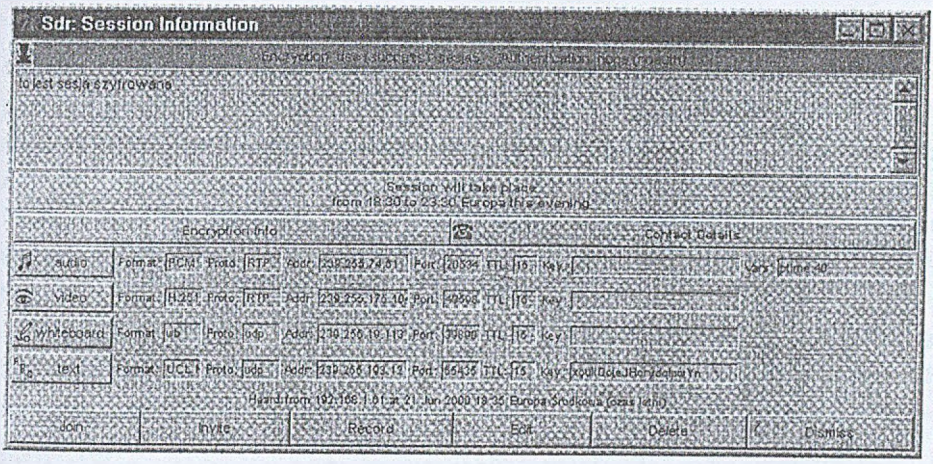
Za pomocą protokołu SIP możliwe jest zestawienie połączenia zarówno w obrębie sieci Internet, jak i za pośrednictwem bramy także z sieciami telefonicznymi czy ISDN. Protokół SIP pozwala na zestawienie połączenia zarówno z wykorzystaniem rezerwacji zasobów, jak i bez rezerwacji zasobów. Jeżeli wykorzystujemy rezerwacje zasobów, SIP dostarcza jedynie informacje do realizacji rezerwacji. Realizacją rezerwacji zasobów zajmują się inne protokoły.

Przedstawione powyżej protokoły zostały zaimplementowane w aplikacjach służących do zarządzania sesjami wielopunktowymi. Przy ich pomocy można stworzyć lub dołączać się do już istniejącej sesji wielopunktowej. Ponadto umożliwiają uruchamianie aplikacji konferencyjnych. Najczęściej stosowane to: **sdr**, **sad** i **sd**.



Rys. 2. Informacje o sesjach wielopunktowych w programie sdr
Fig. 2. Information about multicast session in sdr application

Aplikacja **sdr** jest narzędziem umożliwiającym stworzenie multimedialnej sesji wielopunktowej, a także włączenie się do już istniejącej sesji (rys. 2). Aplikacja **sdr** każdą sesję wielopunktową rozgłasza w sieci. Posiada również możliwość zrealizowania poufnej zapowiedzi konferencji. Oznacza to, iż możliwe jest poinformowanie o danej sesji tylko wyznaczoną grupę osób. Dodatkową możliwością jest także szyfrowanie bieżącej konferencji.



Rys. 3. Parametry multimedialnej sesji wielopunktowej w programie sdr
Fig. 3. Parameters of multimedia multicast session in sdr application

Dla sesji multimedialnej możliwe jest w aplikacji **sdr** zdefiniowanie szeregu parametrów. Przykładowe parametry sesji zostały przedstawione na rysunku 3.

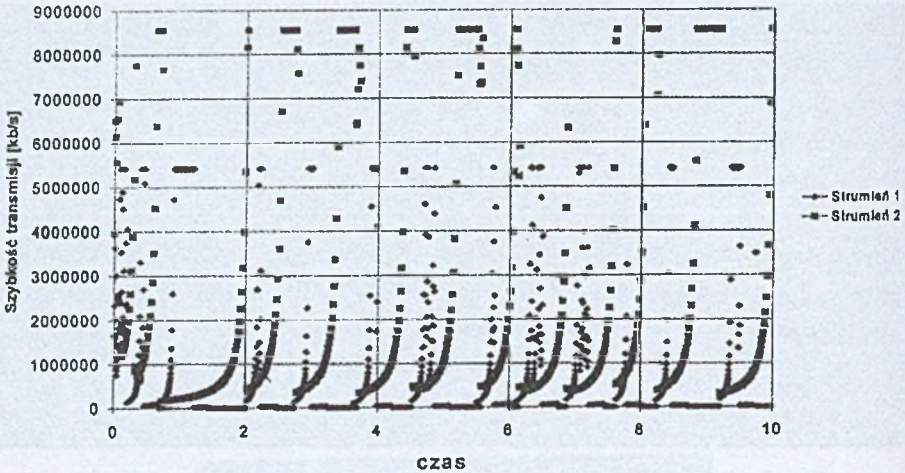
Zarządzaniem transmisją w trakcie trwania sesji zajmują się inne protokoły. W przypadku systemów konferencyjnych (zgodnych z zaleceniami ITU) wykorzystywane są protokoły zdefiniowane w dokumentach H.323 i H.332. W sieci Internet dla multimedialnych aplikacji strumieniowych najczęściej stosuje się protokół RTSP.

Protokół RTSP (*Real Time Streaming Protocol*) jest protokołem poziomu aplikacji (ang. *application-level*) służącym do sterowania dostarczaniem danych transmitowanych w czasie rzeczywistym [11]. RTSP dostarcza elastyczną strukturę umożliwiającą kontrolowanie przesyłanych na żądanie danych czasu rzeczywistego, takich jak audio i video. Podczas sesji RTSP klient RTSP ma możliwość otwierania i zamykania wielu połączeń transportowych do serwera w celu wydania rozkazów RTSP.

Protokół RTSP wspiera następujące operacje:

- opis prezentacji z serwera mediów,
- zaproszenie grupy serwerów mediów do konferencji,
- dodawanie mediów do istniejących prezentacji.

Strumienie sterowane przez RTSP mogą korzystać z RTP, ale działanie RTSP nie zależy od mechanizmu transportowego używanego do przenoszenia ciągłych mediów. Można chociażby użyć protokołu UDP czy TCP. Protokół RTSP jest podobny w składni i operacjach do HTTP 1.1. Jednak protokół RTSP różni się fundamentalnie od HTTP w dostawie danych. HTTP jest asymetrycznym protokołem, gdzie klient żąda emisji, a serwer odpowiedzi. W RTSP zarówno klient media, jak i serwer media może żądać emisji.



Rys. 4. Przykład oddziaływania dwóch strumieni multimedialnych i ich rywalizacja o pasmo transmisyjne
Fig. 4. Interactions between two stream in multimedia transmission

Podczas transmisji multimedialnej w łączu dostępowych następują interakcje pomiędzy strumieniami. Na rysunku 4 przedstawiono przykładowe oddziaływanie dwóch strumieni multimedialnych o zbliżonej charakterystyce ruchowej źródła, transmitowanych z wykorzystaniem RTP. Widoczne jest przejmowanie cykliczne pasma przez obydwa strumienie. Takie zjawisko jest niekorzystne dla efektywnej pracy sieci. Dlatego też stosuje się środki zaradcze. W jednej z metod informacje dostarczane przez RTCP pozwalają na pozyskanie danych o stratach transmitowanych strumieni, jak i o aktualnej przepływności. Informacje te przekazuje się do aplikacji, aby w znacznym stopniu wyrównać zajmowane pasmo i ustabilizować pracę sieci. Inną możliwością jest zastosowanie rezerwacji zasobów.

4. Rezerwowanie zasobów sieci

Ważnym elementem realizacji zarządzania sesjami multimedialnymi jest rezerwacja zasobów. Problem rezerwacji został w znacznym stopniu rozwiązany w technologii ATM. Jest on jednak możliwy do stosowania tylko w sieci homogenicznej ATM. W sieci Internet, bazującej na technologii IP, transmisja odbywa się zgodnie z regułą „best effort". Dlatego też, dla zapewnienia jakości usług (QoS) w sieciach IP, IETF zaproponowało dwie alternatywne architektury sieci IP:

- z rezerwacją zasobów (ang. *Integrated Services*),
- z priorytetami dla określonych strumieni ruchu (ang. *Differentiated Services*).

Architektura *DiffServ* przyporządkowuje strumienie danych do określonych klas usług i są one transmitowane z określonymi priorytetami. Przyporządkowanie następuje na podstawie pola TOS protokołu IPv4 lub pola TC protokołu IPv6. Integralnym elementem tej architektury jest zestaw reguł przekazywania pakietów w routerze tzw. PHB (*Per Hop Behaviors*). W chwili obecnej są realizowane dwie usługi: best effort oraz Premium. Usługa Premium może być traktowana jako emulacja łącza danych o stałej szybkości. Przykładem aplikacji korzystających z usługi Premium są aplikacje konferencyjne audio i wideo oraz transmisja głosu w sieciach IP (ang. *Voice over IP*).

Architektura IntServ realizuje rezerwację dla pojedynczego lub zagregowanego strumienia danych. Aplikacje, które chcą zarezerwować zasoby, muszą poprzez protokół rezerwacji zasobów (np. RSVP) dokonać rezerwacji dla, określonego przez identyfikator, strumienia.

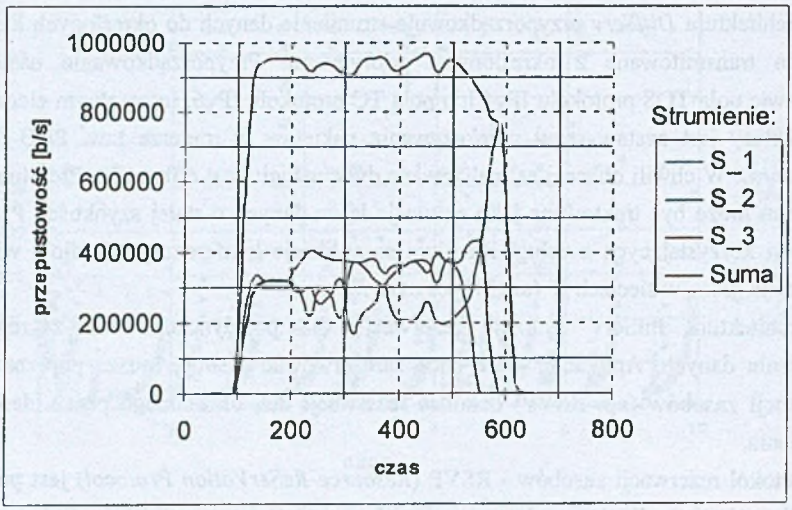
Protokół rezerwacji zasobów - RSVP (*Resource ReSerVation Protocol*) jest protokołem sygnalizacyjnym realizującym dostarczanie zgłoszeń QoS do wszystkich routerów w węzłach wzdłuż ścieżek strumieni oraz ustalenie i utrzymanie stanu obsługi żądań. RSVP współpracuje z IPv4 [7] lub IPv6 [3].

RSVP nie jest protokołem routingu, dlatego nie narzuca specyficznej metodyki znajdowania trasy. Protokół RSVP jest konstrukcją do operowania na bieżących i przyszłych protokołach routingu zarówno w trybie punkt-punk, jak i wielopunktowych. Proces RSVP sprawdza lokalne bazy danych routingu w celu otrzymania trasy.

Protokół ten zgłasza zapotrzebowanie zasobów dla strumieni jednokierunkowych. Proces rezerwacji rozpoczyna się z chwilą, gdy źródło danych, na przykład aplikacja serwera, wysyła do potencjalnego odbiorcy komendę PATH. Komenda zawiera identyfikator strumienia, który wskazuje routerom, jak rozpoznać datagramy jako część sesji RSVP. Mechanizm ten wspierany jest przez protokół IPv6 [3], który posiada identyfikator strumienia. Komenda PATH opisuje strumień informacji, który ma nadejść. Każdy router obsługujący komendę PATH zapamiętuje identyfikator przepływu i łącze, przez które przeszła komenda. Pozwala to routerom rozpoznawać drogę przepływu danych i upewnić się, czy każdy router jest przygotowany do zarezerwowania niezbędnych zasobów.

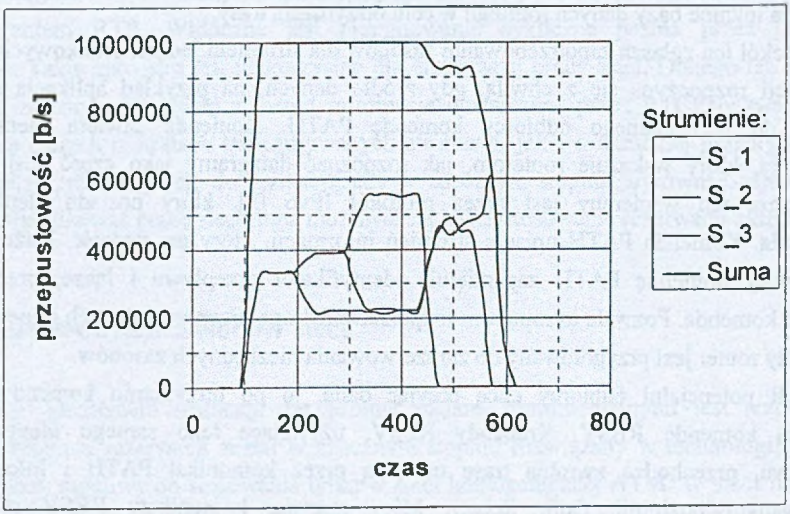
Jeżeli potencjalni odbiorcy chcą przyjąć dane, to po otrzymaniu komendy PATH wysyłają komendę RESV. Komendy RESV, używające tego samego identyfikatora przepływu, przechodzą zwrotną trasę ustaloną przez komunikat PATH i informują o oczekiwanym poziomie QoS. Router, który otrzyma komunikaty RESV od wielu użytkowników, będzie je scalał w pojedynczą komendę.

Jeżeli router może zapewnić wymagany poziom QoS, to zapisze rezerwację do tablicy przepływu. W przeciwnym wypadku router odrzuci rezerwację.



Rys. 5. Optymalne ustawienie buforów w routerach RSVP
Fig. 5. The optimal buffers settings in RSVP routers

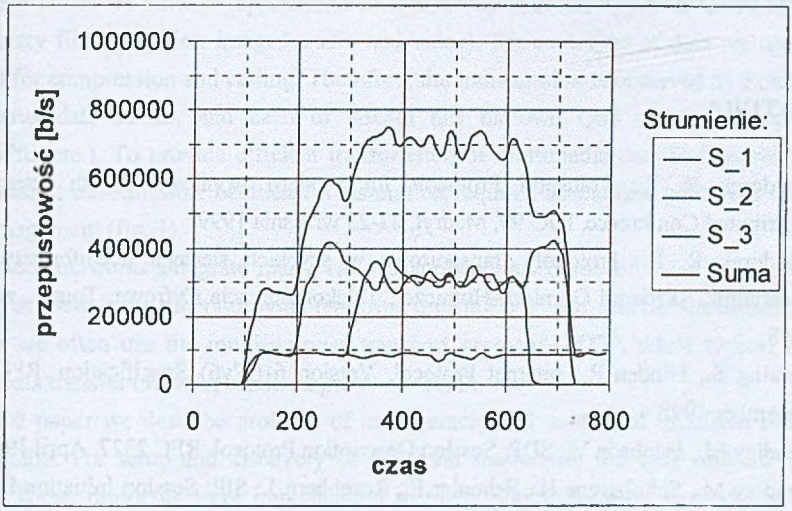
W przypadku transmisji RSVP ważne jest odpowiednie ustalenie buforów w routerach RSVP do każdego rodzaju ruchu. Na rys. 5 przedstawiono zachowanie sieci podczas rezerwacji przy optymalnej wielkości buforów.



Rys. 6. Duże wartości buforów dla ruchu "best effort" w routearch RSVP
Fig. 6. The big value of buffers size for best effort traffic

Na rys. 6 przedstawiono sytuację, w której router posiada bardzo duże bufory dla ruchu typu "best effort". W tym wypadku wykorzystanie sieci jest maksymalne. Wymaga to jednak

dużych zasobów sprzętowych routera (bardzo dużych buforów), a zysk dla aplikacji typu "best effort" jest niewielki.



Rys. 7. Zbyt małe wartości buforów w routerach RSVP
Fig. 7. Too small RSVP buffers in RSVP routers

Ustalenie zbyt małych wartości buforów powoduje (rys. 7), że wykorzystanie dostępnego pasma jest niepełne. Ruch "best effort", jak i podlegający rezerwacji wykazuje znaczne straty, a wartości rezerwacji nie są zachowane.

Dla efektywnej realizacji usługi rezerwacji zasobów istotne jest kompleksowe przeanalizowanie wielu aspektów. Należy rozważyć architekturę sieci i zastosowane mechanizmy. Konieczne jest też wyznaczenie optymalnych wartości buforów dla realizowanych transmisji multimedialnych. Można to zrealizować na drodze symulacji połączonych z analizą realnych sieci.

5. Podsumowanie

W artykule przedstawiono zagadnienia związane ze sterowaniem wielopunktowymi aplikacjami multimedialnymi. Wskazano, jakie protokoły wykorzystywane są do realizacji poszczególnych zadań. Wskazano też na problemy występujące podczas wielopunktowych transmisji multimedialnych.

W przypadku sieci o limitowanych zasobach w wielu przypadkach trzeba stosować rezerwację zasobów, np. za pomocą protokołu RSVP. Wprowadzenie transmisji multimedialnej (korzystającej z protokołu UDP lub RTP) objawia się również degradacją

jakości połączeń TCP. Strumień multimedialny bez kontroli przepływu w skuteczny sposób może stłumić transmisję TCP. Wprowadzając nowe technologie, trzeba zwrócić zatem uwagę, jak będą się zachowywały już istniejące rozwiązania.

LITERATURA

1. Chodorek R. R.: Transport Protocols for Modern Applications, 4th International Distributed Conference, IDC'99, Madryt, 22-23 września 1999.
2. Chodorek R. R.: Protokoły transportowe w szybkich sieciach teleinformatycznych. Kwartalnik Akademii Górniczo-Hutniczej: Telekomunikacja Cyfrowa, Tom 1, zeszyt 4, 1998.
3. Deering S., Hinden R.: Internet Protocol, Version 6 (IPv6) Specification. RFC 2460. December 1998.
4. Handley M., Jacobson V.: SDP: Session Description Protocol. RFC 2327. April 1998.
5. Handley M., Schulzrinne H., Schooler E., Rosenberg J.: SIP: Session Initiation Protocol. RFC 2543. March 1999.
6. Mathis M., Mahdavi J., Floyd S., Romanow A.: TCP Selective Acknowledgement Options. RFC 2018. October 1996.
7. Postel J.: Internet Protocol. RFC 791, USC/Information Sciences Institute, Sepetember 1981.
8. Postel J.: Transmission Control Protocol. RFC 793, USC/Information Sciences Institute. Sepetember 1981.
9. Postel J.: User Datagram Protocol. RFC 768, USC/Information Sciences Institute. Sepetember 1981.
10. Schulzrinne H., Casner S., Frederick R., Jacobson V.: RTP: A Transport Protocol for Real-Time Applications. RFC 1889. January 1996.
11. Schulzrinne H., Rao A., Lanphier R.: Real Time Streaming Protocol (RTSP). RFC 2326. April 1998.
12. Stevens W.: TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC 2001. January 1997.

Recenzent: Dr inż. Bartłomiej Zieliński

Wpłynęło do Redakcji 30 marca 2001 r.

Abstract

The multimedia network application collects many different types of data (e.g. HTML text, binary files, graphics, images, audio and video). For each type of data we use specific method for compression and coding. Therefore, the transmission is observed as a multiplexed continuous data stream and each of stream has its own QoS characteristics (timing, bandwidth, etc.). To provide efficient transmission of multimedia data in Internet we often use multicast transmission. Multicast transmission required specialized protocols for control and management (fig. 1).

Modern networks integrate many various applications. Some of them required reliable data transmission; the others provide real time transmission. For reliable transmission in the Internet we often use the multi-purpose transport protocol - TCP, while typical real-time multimedia transfer utilize specialized protocol - RTP.

In the paper we describe problem of management and control of multicast multimedia transmission. For setup and discovery of multicast session we use SDP and SIP protocol (fig. 1). Those protocols were implemented in many applications for example in *sdr*. On fig. 2 and fig. 3 conference participants and conference parameters are presented. Interaction between multimedia streams (fig. 4) also was analyzed.

The last problem described in the paper is reservation of resources for multimedia transmission. RSVP protocol and its buffer allocation for RSVP reservation (fig. 5,6,7) was described in the paper.