

Piotr Arabas<sup>1,2</sup>, Mariusz Kamola<sup>1,2</sup>

<sup>1</sup> Naukowa i Akademicka Sieć Komputerowa jbr

<sup>2</sup> Instytut Automatyki i Informatyki Stosowanej, Politechnika Warszawska

## Świadczenie usług przesyłowych z gwarancjami jakości w sieci IP – idea i prototyp systemu

W artykule zaproponowano architekturę i przedstawiono wyniki badań prototypu systemu dynamicznej kontraktacji usług przesyłowych w sieci IP. System przyjmuje od klientów operatora telekomunikacyjnego zlecenia zestawienia kanału transmisyjnego w warstwie IP, o określonych parametrach. Naszkicowano podział funkcjonalny systemu i sprecyzowano zadania poszczególnych modułów – również dla scenariusza z interakcją wielu operatorów. Przedstawiono szczegóły implementacji i wyniki testów prototypu takiego systemu, wykorzystującego technologię DiffServ w połączeniu z odpowiednią strategią dopuszczania połączeń i inżynierią tras ruchu. Przeprowadzono szereg eksperymentów weryfikujących wydajność, skuteczność i niezawodność przyjętych rozwiązań. Wyniki potwierdzają możliwość wdrożenia systemu w wersji dopracowanej i kompletnej, w sieci roboczej operatora. Przedstawione prace były częściowo finansowane w ramach Projektu Badawczego Zamawianego PBZ-MNiSW-02/II/2007 (umowa nr PBZ/BUR/2).

### 1. Wstęp

Nieustanny wzrost zapotrzebowania na przepustowość w sieciach IP jest niekwestionowany. Wynika on w dużej mierze ze spadku cen sprzętu umożliwiającego odbiór, nadawanie i tworzenie przekazu audiowizualnego w wysokiej rozdzielczości: wielkogabarytowych telewizorów, wysokowydajnych komputerów służących jako konsole do gier, miniaturowych ale w pełni funkcjonalnych aparatów fotograficznych wbudowanych w telefony komórkowe itp. Użytkownicy – tak indywidualni jak i instytucjonalni – używając aplikacji obciążają sieć przesyłową, przy czym można założyć, że korzystanie z usług oferujących wysoką rozdzielczość przekazu, bardziej absorbującego odbiorcę, odbywa się w sposób planowany. Przykładami takich zachowań niech będą wideokonferencja, oglądanie filmu pełnometrażowego, wspólna gra w sieci.

Racjonalnym rozwiązaniem problemu zapewnienia użytkownikom gwarancji na jakość szerokopasmowych usług przesyłowych (QoS – *quality of service*) realizowanych w sposób planowy byłoby wdrożenie systemu rezerwacji zasobów sieciowych na czas realizacji usługi. System wykorzystywałby obecnie niewykorzystywaną informację, jaką jest deklarowany przez użytkownika czas rozpoczęcia i zakończenia korzystania z usługi, uwzględniając ją w harmonogramie realizacji kontraktów, konfiguracji sieci, wycenie usługi. Autorzy opracowali już koncepcję działania takiego systemu w skali globalnej, skupiając się na teoretycznej analizie kwestii wydajnościowych [4]. Zakładała ona, iż współpracujący operatorzy dysponują odpowiednimi systemami umożliwiającymi dynamiczne zawieranie kontraktów na przesył danych po fragmencie ścieżki transmisyjnej przypadającej w ich własnej sieci. Jednakże systemy takie nie są powszechnie stosowane. Autorzy postanowili więc wykonać własny prototyp takiego systemu, aby oszacować nakład wymaganych środków do jego stworzenia, wykryć ewentualne problemy techniczne, a następnie ocenić działanie prototypu: skuteczność, wydajność, stabilność.

W dalszej części tego dokumentu, w rozdziale 2 został przedstawiony kontekst, w jakim prowadzono prace, tj. aktualnie istniejące bądź opracowywane systemy o podobnej funkcjonalności. Rozdział 3 prezentuje ideę działania i ogólną architekturę proponowanego systemu. Rozdział 4 zawiera opis prototypu systemu. W rozdziale 5 zamieszczono wyniki testów prototypu systemu. Końcowy, 6 rozdział zawiera wnioski oraz przedstawia perspektywy dalszych prac.

## 2. Systemy gwarantujące jakość usług przesyłowych

W ostatnich latach prowadzono na świecie prace mające na celu skonstruowanie systemów pozwalających na dostarczanie usług sieciowych z gwarancjami jakości, oraz na ich kontraktowanie w sposób dynamiczny. Prace te można podzielić na dwa nurty: pierwszy związany z aspektami technologicznymi dostarczania takich usług, oraz drugi zajmujący się głównie kwestiami ekonomicznymi, takimi jak organizacja handlu usługami sieciowymi, wycena usług czy konstrukcja SLA (*service level agreement*). Prace prowadzono zarówno w środowiskach akademickich, przedsiębiorstwach jak i w konsorcjach; charakteryzują się więc one zróżnicowanym nastawieniem do wątków stricte badawczych, praktycznej stosowalności wyników i rzeczywistej możliwości komercjalizacji uzyskanych produktów.

### 2.1 Projekty badawcze

Przeważająca część rezultatów naukowych jest zawdzięczana badaniom prowadzonym w projektach dofinansowywanych z programów ramowych Unii Europejskiej. Pomimo udziału w nich, w zróżnicowanym zakresie, partnerów przemysłowych, utrzymały one charakter badawczy, niejednokrotnie rozwijając idee powstałe w jednym, inicjującym projekt, środowisku naukowym. Na szczególną uwagę zasługują, zajmujące się problematyką technicznej realizacji transmisji z gwarancjami usług, projekty TEQUILA i MESCAL oraz AQUILA i EuQOS.

Główną zasługą projektu TEQUILA stało się zaproponowanie dość zaawansowanej metodyki specyfikacji jakości usług (SLS – *service level specification*), będącej istotną częścią SLA, i wykorzystanej przy definicji protokołu rezerwacji zasobów. Środkiem technicznym dla zapewnienia jakości transmisji było różnicowanie obsługi pakietów za pomocą mechanizmów oferowanych przez DiffServ (*differentiated services*), w połączeniu z dopuszczaniem połączeń (CAC – *connection admission control*) do sieci, możliwym dzięki zastosowaniu wspomnianego protokołu rezerwacji. Wadą opracowanego rozwiązania była słaba skalowalność wynikająca ze zbytnej centralizacji sterowania dostępem do sieci.

Podobne rozwiązanie, tzn. łączące DiffServ i CAC w sieci jednodomenowej, analizował projekt AQUILA. Położono w nim większy nacisk na powiązanie usług sieciowych widzianych przez użytkownika końcowego z dostarczaniem przez sieć gwarancjami jakości oraz na rozproszenie logiki dopuszczania połączeń. Cel osiągnięto stosując architekturę wieloagentową, w której agenty-brokery pasma odpowiadały za negocjacje parametrów transmisji z aplikacją użytkownika.

Jak już zostało wspomniane, wadą rozwiązań opracowanych w obu projektach była słaba skalowalność systemu, wynikająca w dużej mierze z niedostatecznego uwzględnienia problemów powstających na styku wielu oddzielnie zarządzanych domen, jak również brak lub niedostateczne zaawansowanie algorytmów związanych z optymalizacją rozptywu ruchu. Konieczność rozwiązania tych kwestii była w istocie przyczyną kontynuacji wspomnianych projektów: MESCAL stał się bezpośrednią kontynuacją projektu TEQUILA, skupioną na zapewnieniu jakości ruchu przesyłanego między wieloma domenami. Miał temu służyć opracowany mechanizm mapowania klas jakości oraz protokół routingu qBGP, wspomagający międzydomenową inżynierię ruchu.

Podobnie projekt EuQOS, który można postrzegać jako kontynuację niektórych wątków projektu AQUILA, skupiał się na zapewnieniu QoS między domenami, przy czym zaproponowana w nim architektura programowych składników systemu (agentów nadzorujących przydział zasobów, optymalizację tras i konfigurację sprzętu) wydaje się najdojrzalsza, tym bardziej, że wyspecyfikowano protokoły rezerwacji i routingu uwzględniające wymagania QoS, jak również opracowano protokoły komunikacji między elementami systemu. W kontekście omawianych tutaj projektów należy również wspomnieć propozycję systemu RNAP [10], łączącą aspekty techniczne i ekonomiczne, i niewątpliwie inspirującą do dalszych badań.

Jednym z bardziej zaawansowanych, jeśli chodzi o organizację współpracy wielu heterogenicznych domen, wydaje się być projekt NETQOS, którego celem jest stworzenie systemu połączonych, autonomicznych agentów dostarczających usług o gwarantowanej jakości. Rozwiązanie to

wyduje się nowatorskie i daje nadzieję na powstanie systemu niezwykle elastycznie reagującego tak na zapotrzebowania użytkowników, jak i na awarie oraz inne sytuacje nadzwyczajne.

Wśród projektów umiejscowionych w nurcie ekonomicznym szczególne miejsce zajmuje M3I, którego jednym z głównych tematów było modelowanie zależności między podmiotami biorącymi udział w dostarczaniu usług i ekonomicznymi metodami sterowania ruchem (poprzez np. wycenę uwzględniającą dostępność zasobów), a drugim tematem – specyfikacja protokołu rezerwacji zbliżonego do RSVP. Projektem zajmującym się niemalże wyłącznie optymalną wyceną usług sieciowych był QOSIPS, przy czym należy zauważyć, że mimo dość znacznego wyabstrahowania od kwestii związanych z realizacją usług w systemie telekomunikacyjnym, modelowany był wpływ zachowań użytkowników na odczuwany przez nich poziom jakości usług. Znacznie ogólniejszych wniosków dostarczył projekt CoCOMBINE, zorientowany głównie na efekty socjoekonomiczne powszechnego dostępu do internetu, w tym kwestie związane z konkurencją (lub jej brakiem) na rynku telekomunikacyjnym i koniecznością jego regulacji.

## 2.2 Systemy komercyjne

Wśród rozwiązań komercyjnych – o ile wiadomo autorom artykułu – nie istnieje zintegrowany produkt umożliwiający i inżynierię ruchu zorientowaną na realizację kontraktów przesyłowych z zapewnieniem jakości, i wycenę (oraz sprzedaż) tychże usług. Poniżej zostaną przedstawione wdrożone rozwiązania cząstkowe z obu sfer, jednakże operatorzy wystrzegają się ich sprzęgnięcia programowego, utożsamiając je z uprawnieniem klientów do nieograniczonego manipulowania własną infrastrukturą sieciową.

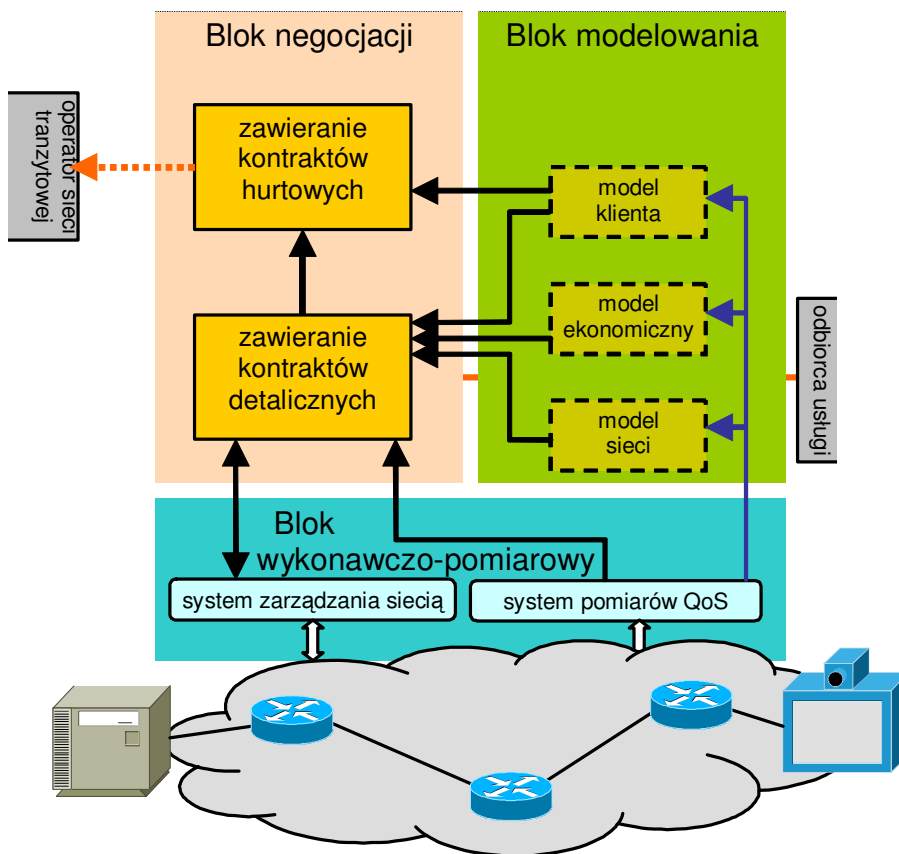
Platformą transakcyjną umożliwiającą kontraktowanie w czasie rzeczywistym usług przesyłowych w Nowym Jorku i okolicach jest MERKATO [7]. Dostawcy przepustowości ubiegają się o zlecenia klientów na np. krótkoterminowe zestawienie kanału transmisyjnego o określonych parametrach. Rolą platformy jest przede wszystkim kojarzenie ofert kupna i sprzedaży, bez specyfikowania mechanizmu wykonawczego zestawiającego usługę. W szczególności może być ona zrealizowana poprzez ręczną konfigurację urządzeń sieciowych operatora otrzymującego zlecenie (takie też są realia zestawiania kontraktów u operatorów – zarówno małych, jak i dużych).

Z drugiej strony, istnieją technologie i towarzyszące im systemy wygodnego zarządzania kanałami transmisyjnymi w sieciach IP, również rozległych. Taką technologią jest MPLS (*multiprotocol label switching*), który w połączeniu z DiffServ jako mechanizmem różnicowania jakości usług oraz RSVP (*resource reservation protocol*) jako protokołem sygnalizacyjnym umożliwia wygodne zestawienie kanału transmisyjnego o zadanych parametrach. Żądanie zestawienia kanału jest propagowane kaskadowo pomiędzy routerami, zwalniając administratora z obowiązku konfigurowania każdego z urządzeń oddzielnie. Ścieżki powstałe w ten sposób można traktować jako pełnowartościowe łącza wirtualne.

Korzystanie z tuneli MPLS jest jeszcze wygodniejsze dzięki kompletnym, zaawansowanym systemom zarządzania siecią, jak np. IP Solution Center firmy Cisco [3]. Umożliwiają one nie tylko zarządzanie operacyjne zasobami sieciowymi w zunifikowany sposób, z silnym ukierunkowaniem na zapewnienie niezawodności działania – ale również dostarczają mechanizmów optymalizacji działania sieci, np. tras routingu. IP Solution Center oferuje nowoczesne API, umożliwiając tym samym sprzęg z aplikacjami zewnętrznymi. Lecz – jak już powiedzieliśmy – nie jest on wykorzystywany do automatyzacji procesu zawierania kontraktów na usługi przesyłowe.

## 3. Funkcjonalność i architektura proponowanego systemu

Narzędzia przedstawione w poprzednim rozdziale nie stały się bodźcem do rozpowszechnienia systemów rezerwacji zasobów sieciowych z różnych powodów. Produkty będące wynikiem projektów naukowo-badawczych z reguły pozostają w fazie koncepcji lub – co najwyżej – makiety albo prototypu. Z kolei do stosowania produktów komercyjnych operatorzy odnoszą się z wielką niechęcią gdyż, oprócz obawy przed nadawaniem prerogatyw użytkownikom sieci, koszt ich wdrożenia jest



Rys. 1. Architektura systemu rezerwacji zasobów sieciowych

nieproporcjonalnie wysoki, zważywszy znaczne ryzyko inwestycyjne. Produkty komercyjne są po prostu za drogie. W takiej sytuacji opracowanie alternatywnego rozwiązania zawierającego zarówno system transakcyjny, jak i moduł wykonawczy dla sieci IP, ma sens badawczy i jednocześnie otwiera perspektywę udanej komercjalizacji.

Głównym zadaniem proponowanego systemu rezerwacji zasobów sieciowych jest umożliwienie użytkownikowi zawarcia kontraktu na usługę przesyłową o określonych parametrach: czasie trwania, punktach końcowych transmisji, jakości usług, cenie. Usługa jest realizowana w obrębie sieci operatora, aczkolwiek jej świadczenie w skali globalnej, w drodze współpracy wielu operatorów, jest również realne po uzgodnieniu punktów końcowych transmisji, czasu trwania i parametrów jakościowych.

Założono, że system negocjacji i realizacji kontraktów w obrębie sieci jednego operatora będzie miał architekturę scentralizowaną. System jest wówczas łatwiejszy w zaprojektowaniu, wykonaniu i zarządzaniu, narzuca mniejsze wymagania na sposób interakcji z użytkownikiem i z urządzeniami sieciowymi. Jednocześnie, dzięki hierarchicznej organizacji podobnych systemów w sieci rozległej, nie ma powodu obawiać się o skalowalność całego rozwiązania.

System rezerwacji zasobów dzieli się na bloki funkcjonalne złożone z modułów, wg Rys. 1. Blok negocjacji stanowi centralny element systemu: tu zachodzi bezpośrednia interakcja między systemem a użytkownikami oraz operatorami hurtowymi ulokowanymi wyżej w hierarchii. Bloki modelowania i wykonawczo-pomiarowy pełnią rolę służebną wobec bloku negocjacji, umożliwiając, odpowiednio, właściwe funkcjonowanie logiki aplikacji oraz realizację kontraktu.

### 3.1 Blok negocjacji

Blok ten jako jedyny wchodzi w interakcję z **użytkownikami detalicznymi**, przyjmując bądź odrzucając ich zlecenia zawarcia kontraktu. Decyzja o zrealizowaniu kontraktu jest uwarunkowana tylko dwoma czynnikami:

- możliwościami technicznymi realizacji kontraktu,
- opłacalnością realizacji kontraktu.

O zaistnieniu możliwości technicznych przesądza stan sieci w czasie obowiązywania kontraktu. Jako że kontrakt jest w ogólności zawierany z wyprzedzeniem, możliwości techniczne jego realizacji mogą być jedynie prognozowane, i zależeć od:

- dostępnej infrastruktury sieciowej,
- innych zawartych kontraktów

w okresie, w jakim ma obowiązywać nowy kontrakt. Czynniki te są modelowane odpowiednio przez modele sieci i klienta, zawarte w bloku modelowania. Modelowanie infrastruktury sieciowej nie nastręcza trudności, gdyż jest ona z reguły wolnozmienna. Natomiast modelowanie innych kontraktów jest zadaniem skomplikowanym, ponieważ powinno uwzględniać zarówno parametry formalne kontraktu (np. przepustowość i opóźnienie), jak również prognozowane *faktyczne* wykorzystanie sieci przez użytkownika.

Definicja opłacalności pojedynczego kontraktu uwzględniająca więcej niż tylko prosty bilans zysków odzwierciedla długofalową politykę operatora. I tak np. wdrażając politykę podnoszenia marży w miarę malenia dostępnej niewykorzystanej przepustowości, operator dokonuje dyskryminacji cenowej, zapewniając dostępność usługi dla klientów hojniejszych i skłaniając mniej zasobnych klientów do zawierania umów z dużym wyprzedzeniem. Inaczej ci ostatni mieliby małe szanse na zawarcie kontraktu, co skutkowałoby frustracją, a następnie – rezygnacją z usługi. Z drugiej strony, operator ponosząc głównie koszty stałe własnej infrastruktury, powinien zabiegać o stymulowanie popytu, by sieć ciągle była wykorzystywana.

Oszacowanie opłacalności kontraktu staje się jeszcze bardziej złożone, jeśli operator ma możliwość dynamicznego zawierania **kontraktów hurtowych** na przepustowość na wyjściach ze swojej sieci. Wówczas zasoby służące ruchowi tranzytowemu nie są już niezmiennie w czasie; pojawia się możliwość ich okresowej renegotjacji. Taka renegotjacja wynika z analizy harmonogramu obowiązywania kontraktów – ale też strategia (a więc i wynikający z niej harmonogram) zawierania kontraktów zależy od kosztów kontraktów tranzytowych. Pojawia się więc wzajemna zależność, która w proponowanym systemie może zostać usunięta przez założenie, że okresy obowiązywania kontraktów hurtowych są dłuższe niż horyzont, na którym podejmowane są decyzje operacyjne dotyczące kontraktów detalicznych. Wobec tego, kontrakty hurtowe są postrzegane w module zawierania kontraktów detalicznych jako niezmiennie. Moduł ten podejmuje decyzje posiłkując się wyłącznie modelem ekonomicznym. Natomiast moduł kontraktów hurtowych działa w sposób autonomiczny, analizując *post factum* kontrakty detaliczne (przyjęte oraz odrzucone) i, w przypadku wykrycia możliwości poprawy uzyskanego rozwiązania, renegotjuje z operatorem na wyższym szczeblu hierarchii nowy kontrakt, adekwatny do potrzeb. W ten sposób, poprzez rozprzęgnięcie decyzji dotyczących kontraktów hurtowych i detalicznych, uzyskana zostaje skalowalność systemu w skali globalnej.

### 3.2 Blok modelowania

Trzy modele specjalizowane: sieci, ekonomiczny i klienta składają się na blok modelowania, umożliwiając algorytmom negocjacji uzyskiwanie prognoz i – co za tym idzie – realizację optymalnej strategii zawierania kontraktów. Do strojenia modeli sieci i klienta wykorzystywane są głównie dane o kontraktach oraz dane ruchowe, dostępne w obfitości z sieci. Natomiast model ekonomiczny ma głównie charakter ekspercki i nie podlega tak częstym modyfikacjom.

Zadaniem **modelu sieci** jest określenie warunków panujących w sieci podczas realizacji pewnego zbioru kontraktów. Kontrakt  $C_i$  jest opisany zestawem następujących parametrów:

- czasów rozpoczęcia  $\alpha_i$  i zakończenia  $\omega_i$ ,
- punktów początkowego  $o_i$  i końcowego  $t_i$  transmisji,
- przepływności strumienia średniej  $b_i$  i chwilowej  $p_i$ ,
- maksymalnego opóźnienia  $d_i$ , i jego wariancji  $j_i$ ,
- odsetka traconych pakietów  $r_i$ ,

- wektora parametrów  $f_i$  opisujących przewidywany *faktyczny* sposób korzystania przez użytkownika z kontraktu.

Model sieci określa, jak zostaną skonfigurowane urządzenia sieciowe, aby zrealizować określony zestaw kontraktów w sposób optymalny. Model sieci zawiera więc (lub powieła, w zależności od szczegółów konstrukcji systemu) logikę zarządzania siecią. Wyjściem modelu są wielkości opisujące warunki na poszczególnych łączach oraz przewidywane parametry jakościowe dla poszczególnych kontraktów. Przewidywane parametry jakościowe dla kontraktów dotyczą przepływności, opóźnienia i jego wariancji, odsetka traconych pakietów oraz, dodatkowo, trasy realizacji kontraktu (łączy składających się na ścieżkę transmisji).

Przewidywane parametry jakościowe w odniesieniu do łącza są ściśle związane z technologią zapewnienia jakości w sieci IP. Dla zastosowanej w rozwiązaniu prototypowym technologii DiffServ sprowadzają się one do charakterystyk ruchu w każdej z klas i priorytetów DiffServ, wyrażonych liczbowo analogicznie jak dla poszczególnych przepływów.

Dla określenia warunków panujących w sieci w przyszłości, model sieci przechowuje informacje o strukturze sieci, sposobie realizacji kontraktów oraz łącznym wpływie zestawionych kontraktów na wykorzystanie zasobów sieciowych. Te trzy elementy, wraz z zadaniem zbioru kontraktów do realizacji definiują zadanie opracowania optymalnej strategii wykorzystania zasobów sieciowych, któremu topologia sieci oraz kontrakty wraz z formułą mieszania ruchu narzucają ograniczenia. Np. przy realizacji kontraktów z użyciem DiffServ, wynikowa strategia realizacji kontraktów determinuje przynależność każdego z nich do poszczególnych klas ruchu oraz określa sposób gospodarowania wolnymi zasobami – w tym przykładzie może ono polegać na równoważeniu obciążeń wszystkich łączy, albo też na kierowaniu ruchu najkrótszą ścieżką, itp.

Sposób modelowania interakcji pomiędzy strumieniami danych w łączu ściśle zależy od natury i liczby tych strumieni. Najbardziej wyrafinowane podejście polega na zastosowaniu teorii pasma efektywnego [5]; wymaga ono jednak dokładnej znajomości parametrów mieszanego ruchu i znajduje zastosowanie przy transmisji wielu strumieni jednocześnie. Dla mniejszej liczby strumieni można posługiwać się prostszymi technikami modelowania – bardziej pesymistycznymi, albo wręcz symulować działanie sieci (np. ogólnodostępnym symulatorem ns2 [8]). Znaczny czas trwania symulacji raczej wyklucza posługiwanie się nią przy zestawianiu kontraktów w czasie rzeczywistym: może być ona jednak wykorzystana do cyklicznej inżynierii ruchu, polegającej na optymalizacji i rekonfiguracji tras wszystkich aktualnych kontraktów.

Należy zaznaczyć, iż modelowanie sieci z reguły służy weryfikacji możliwości dodania nowego kontraktu do listy już zawartych: wówczas dodatkową informacją przekazywaną modelowi sieci jest lista obowiązujących kontraktów wraz z zestawionymi ścieżkami przesyłu danych. Stanowi ona dodatkowe ograniczenie w zadaniu optymalizacji rozwiązywanym przez model sieci.

Zadaniem **modelu klienta** jest modelowanie indywidualnych zachowań klientów na podstawie danych o zawartych kontraktach, w konfrontacji z faktycznym korzystaniem przez użytkowników z sieci w czasie trwania kontraktu. Zadaniem modelu jest prognozowanie profilu ruchu klienta dla zawieranego kontraktu. Profil ruchu jest następnie wykorzystywany przez model sieci do określenia faktycznych – a nie wynikających tylko z nominalnych parametrów kontraktu, żądanych przez użytkownika – zasobów potrzebnych do realizacji kontraktu. Dzięki takiemu modelowaniu inżynieria ruchu jest wydajna, bo opiera się na rzeczywistym, a nie deklarowanym wykorzystaniu sieci.

Relacja pomiędzy deklarowanymi a faktycznymi potrzebami użytkownika sieci nie jest stała: zależy przeznaczenia kontraktu (tj. rodzaju wykorzystującej go aplikacji). Może również zależeć od ceny tego kontraktu, wg zasady, że klient skwapliwie wykorzysta każdy zakontraktowany kb/s, jeśli nabył go bardzo drogo. Ta zależność wprowadza, niestety, sprzężenie pomiędzy decyzją o możliwości zawarcia kontraktu, a późniejszą decyzją o jego wycenie. Można je zaniedbać w fazie sprawdzania możliwości realizacji kontraktu, przyjmując najbardziej pesymistyczne spośród danych historycznych, dotyczące wykorzystania pasma zarezerwowanego w kontrakcie. To wyjściowe oszacowanie może następnie, w trakcie trwania kontraktu, podlegać weryfikacji na podstawie aktualnych statystyk ruchowych, wpływając na bieżącą strategię realizacji nowych kontraktów.

Zadanie **modelu ekonomicznego** polega na modelowaniu wszelkich aspektów związanych z kosztami i przychodami wynikającymi ze świadczenia usług. Ponieważ na koszt konkretnej usługi wpływają koszty zarówno bezpośrednio z nią związane, jak i związane z usługami składowymi, z których usługa ta korzysta, niezwykle istotne staje się zadanie właściwego podziału kosztów wspólnych. Sprawiedliwy podział kosztów można definiować na różne sposoby [2]; w teorii powinien on eliminować nieuzasadnione przenoszenie kosztów przynależnych do jednej usługi na inną usługę, umożliwiając konkurencyjnym operatorom zaoferowanie usługi bez takiego obciążenia, a więc po niższej cenie. W praktyce, uwzględniając specyfikę rynku, tj. nieliniowy związek cen usług z faktycznymi kosztami, algorytmy podziału kosztów służą zgrubnemu oszacowaniu opłacalności poszczególnych produktów operatora [9]. W każdym przypadku modelowanie kosztów pozostaje bardzo istotnym elementem strategii sprzedaży, albo będąc istotnym składnikiem funkcji celu, albo definiując ograniczenia na ceny-zmienne decyzyjne.

Jak zostało wspomniane w opisie modelu sieci, zasoby wynikające z kontraktów operatora z operatorami hurtowymi położonymi wyżej w hierarchii podlegają okresowym zmianom, wpływając na koszty świadczonych usług. Przyjmujemy jednak konsekwentnie, iż zmiany te zachodzą w zupełnie innej, dłuższej skali czasowej; natomiast podstawowa działalność operacyjna systemu rozgrywa się dla ustalonego zbioru aktywnych kontraktów hurtowych.

Model ekonomiczny opisuje również oddziaływanie cen usług na popyt ze strony klientów. Można w tym celu wykorzystać uznane modele matematyczne, np. Cobba-Douglasa [6], wykorzystujący pojęcie elastyczności do przewidzenia procentowej zmiany w wolumenie sprzedaży (bądź wykorzystania) danej usługi, dla określonej, procentowej zmiany ceny usługi. Z reguły modelowania takiego dokonuje się po wstępnej segmentacji rynku. Parametry modelu uaktualniane są na podstawie obserwowanych reakcji klientów w poszczególnych segmentach, w odpowiedzi na zmiany cen usług. W przypadku dynamicznej kontraktacji usług, ilość danych wykorzystywanych do strojenia modelu jest wystarczająca, ale poprawne funkcjonowanie modelowania i tak w dużym stopniu będzie zależęć od doboru początkowych parametrów i właściwej segmentacji – stąd wcześniejsze stwierdzenie, że model ten ma charakter ekspercki.

Dane wyjściowe modelu ekonomicznego są wykorzystywane w bloku negocjacji zarówno przy zawieraniu kontraktów detalicznych, jak i hurtowych. Zawsze determinują funkcję celu, niezależnie od sposobu jej zdefiniowania; definicja celu może bowiem akcentować udział innych czynników, jak np. wolumenu transmisji (stymulując określone nawyki klientów), liczby niezrealizowanych kontraktów (zmniejszając niezadowolenie klientów) czy odsetka kontraktów niespełniających oczekiwań jakościowych klienta (j.w.). Jednak czynniki te nieodmiennie wyrażają pośrednio potrzebę realizacji takich samych celów ekonomicznych, lecz na dłuższym horyzoncie czasowym.

### 3.3 *Blok wykonawczo-pomiarowy*

Blok ten pełni rolę warstwy adaptacyjnej pomiędzy blokami negocjacji i modelowania, a konkretnymi urządzeniami w sieci teletransmisyjnej operatora. Działanie technologii realizacji kontraktów z gwarancjami jakości stanowi, co prawda, składnik funkcjonowania logiki bloku negocjacji; jednakże z uwagi na różnorodność dostępnych urządzeń oraz sposobów komunikacji niezbędny jest odpowiedni zestaw wyspecjalizowanych narzędzi do komunikowania się ze sprzętem. Zawiera je właśnie blok wykonawczo pomiarowy. Z tych samych powodów zawiera on również narzędzia monitorowania stanu sieci specyficzne dla użytego sprzętu i technologii pomiaru jakości.

Mechanizmy wykonawcze zawierania kontraktu tworzą **podsystem zarządzania siecią**. Dla realizacji całego systemu za pomocą DiffServ łącznie z MPLS, podstawowe funkcje podsystemu polegają na zarządzaniu:

- kryteriami klasyfikacji przepływów,
- egzekwowaniem zakontraktowanych parametrów ruchu dla przepływów na brzegu sieci (*shaping, profiling*),
- priorytetyzacją ruchu poprzez kolejki klasowe związane z interfejsami wyjściowymi routerów (*priority queues*),

- ścieżkami transmisji danych MPLS.

Zarządzanie urządzeniami sieciowymi można zrealizować za pomocą standardowych protokołów, jak SNMP czy LDAP, lub poprzez sesję programu telnet. Urządzenia przeważnie obsługują oba sposoby komunikacji, przy czym nie jest regułą, aby zarządzanie w sesji telnet było mniej wydajne czy skuteczne niż zarządzanie poprzez specjalizowany protokół. Główna trudność przy automatyzacji zarządzania konfiguracją urządzeń sieciowych, niezależnie od sposobu komunikacji, polega jednak na niemożności upewnienia się, czy aktualna konfiguracja urządzenia jest poprawna. W wielu przypadkach można skutecznie zapewnić semantyczną poprawność poszczególnych poleceń, podczas gdy cała ich sekwencja prowadzi do nieprawidłowego funkcjonowania lub całkowitej dysfunkcji urządzenia.

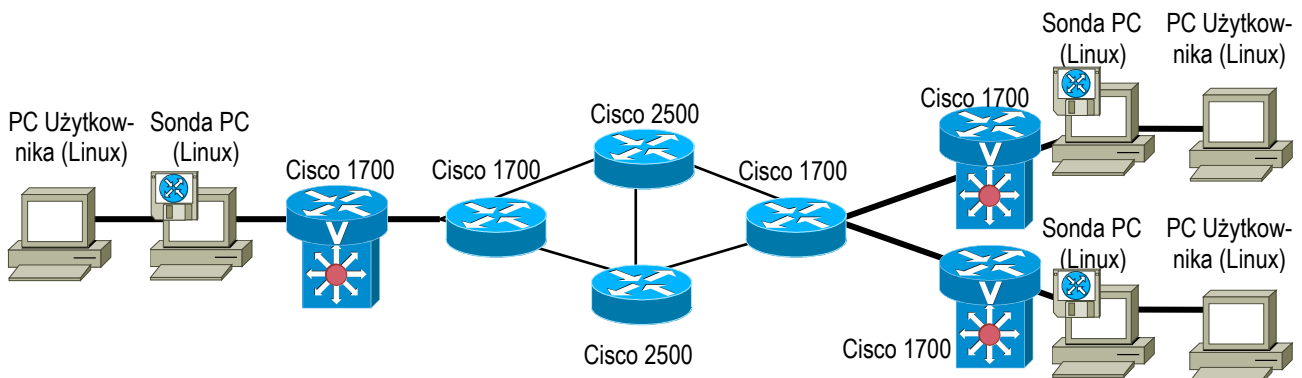
Zagadnienie **pomiaru jakości** świadczonych usług może być rozwiązane na wiele sposobów, angażując w różnym stopniu istniejące oraz dedykowane urządzenia, oprogramowanie oraz same zasoby transmisyjne sieci. Standardowe możliwości monitorowania, udostępniane przez routery, są skromne; można je rozszerzyć uruchamiając na istniejącym sprzęcie dodatkowe moduły programowe, co nie pozostaje bez wpływu na wydajność i stabilność sprzętu. Alternatywą jest umieszczenie w sieci sond monitorujących jakość w sposób pasywny lub aktywny. Takie rozwiązanie jest skuteczne, lecz dość kosztowne. Rozwiązania kompromisowe polegają na umieszczaniu sond w strategicznych punktach sieci, a następnie estymacji parametrów łączy bezpośrednio nie monitorowanych [1].

## 4. Prototyp systemu

W NASK wykonano, a następnie uruchomiono w warunkach laboratoryjnych, prototyp systemu oferujący niepełny zestaw operacji tak, aby móc zweryfikować poprawność i skuteczność działania produktu w wersji załączkowej. Z przyczyn naturalnych realizację prac rozpoczęto od stworzenia sieci testowej, następnie – elementów bloku wykonawczo-pomiarowego, wreszcie – logiki aplikacji.

### 4.1 Sieć testowa i mechanizmy zapewnienia jakości usług

Zestawiono zostało stanowisko testowe złożone z siedmiu routerów Cisco i sześciu komputerów klasy PC. Ideą determinującą wybór topologii połączeń było odwzorowanie minimalnej konfiguracji typowej sieci dostawcy usług w układzie podwójnej gwiazdy z redundantnym rdzeniem, składającej się z dwóch routerów rdzeniowych (podstawowy i zapasowy), routerów brzegowych (z połączeniami do obu routerów rdzeniowych) oraz routerów abonenckich. Dla w miarę kompletnego odwzorowania problemów zachodzących w rzeczywistej sieci przy minimalnej liczbie użytych urządzeń przyjęto, iż sieć powinna posiadać dwa routery brzegowe i trzy abonenckie (router brzegowy, do którego podłączone są dwa routery abonenckie, jest miejscem potencjalnego spiętrzenia ruchu). Konsekwentne wykorzystanie sprzętu konkretnego producenta ułatwia wykonanie procedur komunikacji; odzwierciedla ono również duży udział urządzeń tej marki w sieci roboczej NASK.



Rys. 2. Topologia połączeń stanowiska testowego



Schemat wykonanych połączeń przedstawia Rys. 2. Wykorzystano routery serii 2500 (rdzeń sieci) i 1700 (brzegowe i aboneckie). Aby móc skorzystać z podstawowych możliwości architektury DiffServ (kolejki klasowe, klasyfikacja ruchu za pomocą list dostępu, *shaping* i *policing* oraz zbieranie statystyk kolejek klasowych przez SNMP), konieczne stało się uaktualnienie systemów operacyjnych w stopniu, na jaki pozwalał ograniczony rozmiar pamięci RAM w użytych urządzeniach.

Oprócz routerów, w skład stanowiska testowego wchodziły komputery PC pełniące funkcję komputerów użytkowników, służące do generowania ruchu, oraz komputery pełniące rolę próbników i służące do zbierania danych. Zarówno próbniki, jak i komputery użytkownika połączone są dodatkową siecią w układzie gwiazdowym, wykorzystywaną do zadań zarządczych, organizujących eksperymenty (sieć ta nie jest pokazana na Rys. 2). Łącza pomiędzy routerami zrealizowano stosując interfejsy szeregowe, co pozwala na dość elastyczne wybieranie przepustowości poprzez zmianę częstotliwości taktowania interfejsów (maksymalna osiągalna prędkość to 2 Mb/s).

Przy wszelkich analogiach należy jednak pamiętać o ograniczeniach wynikających z konstrukcji stanowiska testowego, a związanych tak z jego uproszczoną topologią (niewielka liczba węzłów, skupiony charakter stanowiska laboratoryjnego), jak również stosunkowo niewielkimi możliwościami użytego sprzętu (w sensie wydajności, ale również nieimplementowania niektórych bardziej zaawansowanych mechanizmów). Z tego powodu sieć testową należy traktować jako "model w zmniejszonej skali" rzeczywistej sieci. Podczas eksperymentów należy używać przepływów o mniejszych niż spotykane w sieci roboczej, prędkościach transmisji.

W projektowaniu eksperymentów należy kierować się celem eksperymentu i, niestety, często wybierać rozwiązania kompromisowe – np. dla łącza 2 Mb/s niemożliwe jest zbadanie transmisji wielu strumieni wideo o wysokiej jakości. Możliwe jest natomiast przesłanie jednego strumienia wideo o w miarę wysokiej jakości (1–2 Mb/s), lub kilku o jakości niskiej (np. 512 kb/s). Analogicznie, analizując opóźnienia należy zdawać sobie sprawę, że z jednej strony opóźnienia kolejkowe będą większe niż w rzeczywistych sieciach szerokopasmowych (czyli wystąpi przeskalowanie wynikające z niskiej przepustowości łącz); z drugiej zaś strony opóźnienia transmisyjne będą praktycznie zerowe, nieodzwierciedlające zjawisk obserwowanych w rzeczywistych sieciach rozległych.

#### **4.2 Elementy bloku wykonawczo-pomiarowego**

W celu umożliwienia sterowania i monitorowania sprzętu sieciowego, niezbędnym stało się stworzenie biblioteki procedur komunikacyjnych stanowiącej dolną warstwę oprogramowania systemu, realizujących najniższą warstwę abstrakcji. Na podstawie wcześniejszych testów oraz doświadczeń zebranych przez podobne projekty (np. AQUILA), do komunikacji z routerami wybrano aplikację telnet ze względu na jej powszechną dostępność w urządzeniach sieciowych, prostotę oraz dostateczną wydajność i niezawodność.

Ze względu na konieczność wydajnego przetwarzania znacznej ilości informacji tekstowych, do implementacji procedur wybrano język PERL, którego wielką zaletą w tym zastosowaniu jest możliwość wykorzystania gotowych pakietów do komunikacji z routerami poprzez telnet. Wykonana wersja biblioteki została ukierunkowana na komunikację z routerami Cisco i podejmowanie czynności związanych z konfiguracją kontraktów dynamicznych oraz zbieraniem statystyk. Starano się jednakże zaimplementować ją w sposób możliwie elastyczny, pozwalający w przyszłości na jej rozszerzenie o dodatkową funkcjonalność (w sensie np. konfigurowania dodatkowych ustawień urządzeń) jak również przystosowanie do współpracy ze sprzętem innego producenta. Przez wzgląd na to algorytmy rozbito na procedury niskiego poziomu, ściśle zależne od sprzętu (są to w szczególności wszystkie funkcje związane z wydawaniem routerom poleceń systemu operacyjnego), oraz procedury wyższego poziomu, operujące na danych przygotowanych przez procedury niskopoziomowe. Wprowadzenie takiej hierarchii powinno uprościć przygotowanie wersji biblioteki obsługującej sprzęt innych producentów.

Dwa zasadnicze wymagania stawiane tego typu bibliotekom komunikacyjnym, wydajność i niezawodność działania, pozostają niestety w dość wyraźnej sprzeczności. Głównymi kwestiami determinującymi niezawodność są skuteczne nawiązywanie połączenia oraz weryfikacja poprawno-

ści wydawanych poleceń i skuteczności ich wykonania przez urządzenie. Należy pamiętać, że problemy z komunikacją i konfiguracją nie muszą być wcale związane z zaistniałą jawną sytuacją awaryjną (w sensie np. niesprawności łącza czy samego routera) lecz np. ze znacznym obciążeniem urządzenia, skutkującym odroczonej lub w inny sposób wadliwą realizacją poleceń konfiguracyjnych. Wprawdzie większość procedur wykorzystywanej biblioteki informuje o statusie wykonywanej komendy; należy jednak zdawać sobie sprawę, że taka kontrola nie pozwala na wykrycie wszystkich błędów: pozwala wykryć tylko stosunkowo poważne błędy, głównie natury składniowej. Błędy zaistniałe bez objawów na konsoli (np. powołanie się na nieważny numer listy uprawnień czy pominięcie wykonania polecenia przez router) można wykryć tylko na drodze dość szczegółowej analizy stanu routera, obejmującej m.in. badanie zależności pomiędzy aktualnymi wartościami parametrów.

Konieczność prowadzenia takiej analizy powoduje, że rośnie liczba faktycznie wykonywanych poleceń w urządzeniach,<sup>1</sup> zmniejszając wydajność całego systemu. Zastosowanym rozwiązaniem kompromisowym jest okresowe sprawdzanie poprawności konfiguracji polegające na porównywaniu raportowanej przez router konfiguracji ze stanem kontraktów zapisanych w bazach systemu. Podstawowe procedury pozwalające na opisywaną weryfikację zostały przygotowane i wchodzi w skład biblioteki.

Biblioteka komunikacyjna składa się z szeregu skryptów języka Perl. Trzy z nich są wykorzystywane bezpośrednio przez moduł zarządcy do komunikacji z routerami. Zarządzanie i monitorowanie sieci poprzez sekwencję wywołań programów zewnętrznych zastosowano uwagi na prostotę i niezawodność. Funkcjonalność pierwszego skryptu obejmuje:

- wyświetlanie konfiguracji routerów,
- wykonywanie poleceń konfiguracyjnych zawartych w pliku,
- przestawianie prędkości taktowania łączy szeregowych,
- wyświetlanie stanu interfejsów routerów,
- zakładanie i usuwanie list dostępu,
- wyświetlanie ustawień filtrów dyscyplinujących ruch (*policers*),
- wyświetlanie ustawień kolejek,
- konfigurowanie podkolejki DiffServ AF (*assured forwarding*),
- konfigurowanie mapy routingowej.

Drugi skrypt umożliwia kompleksowe konfigurowanie kontraktów. Funkcjonalność skryptu obejmuje:

- założenie kontraktu wraz z filtrem dyscyplinującym (tworzona jest lista dostępu, podkolejka AF oraz filtr),
- założenie kontraktu jak wyżej z wyszukaniem wolnego numeru listy dostępu,
- założenie kontraktu jw. z wyszukaniem wolnego numeru listy dostępu, ustawieniem typowych parametrów zmienności ruchu i sprawdzeniem ustawień filtra dyscyplinującego,
- skasowanie kontraktu,
- założenie kontraktu z filtrem kształtującym ruch (*shaper*).

Trzeci skrypt został przygotowany w celu umożliwienia odczytywania podstawowych informacji o ruchu w sieci, tj.:

- pasma wszystkich kolejek interfejsu,
- statystyk filtrów kształtujących ruch,
- statystyk interfejsu,
- pasma w podziale na kontrakty (zbiorczo) i w pozostałych klasach DiffServ.

---

<sup>1</sup> Dla przykładu, po założeniu listy dostępu poleceniem `access-list permit` konieczne jest opuszczenie trybu konfiguracyjnego i wykonanie polecenia `show access-lists` – czyli wykonanie dwóch poleceń, z których drugie może zwrócić długą listę wszystkich list dostępu, wprowadzając dodatkowe opóźnienie.

### 4.3 Logika negocjacji i realizacji kontraktów

W obecnie zaimplementowanym i testowanym algorytmie negocjacji i realizacji kontraktów zastosowano szereg uproszczeń w stosunku do projektowanej pełnej funkcjonalności, opisanej w rozdz. 3.1. Zrezygnowano z implementacji rdzenia systemu w postaci nieustannie działającej usługi na rzecz programu wsadowego realizującego obsługę żądania zawarcia pojedynczego kontraktu wyszczególnionego w parametrach wywołania. Aktualny stan systemu, będący podstawą decyzji o zestawieniu kontraktu, czyli

- obowiązujące kontrakty,
- bieżące statystyki ruchowe,
- topologia i ścieżki transmisji danych –

jest przechowywany w plikach tekstowych. Klasy realizujące sprzęg z plikami zostały zaprojektowane tak, aby umożliwić późniejszą łatwą integrację z relacyjną bazą danych.

Ze względu na przyjęte uproszczenia program musi symulować działanie docelowego systemu rezerwacji mimo wsadowego charakteru pracy. Wiąże się to, po pierwsze, z bieżącą obsługą żądań użytkownika oraz, po drugie, z kasowaniem wygasłych kontraktów. W pierwszym przypadku realizowany jest opisany poniżej algorytm dopuszczania połączeń, w drugim – algorytm porządkowania bazy kontraktów i konfiguracji sieci. Program jest w obecnej postaci uruchamiany cyklicznie, co jest oczywistym uproszczeniem, dopuszczalnym jednak w tymczasowej implementacji (aktywne przeglądanie listy kontraktów angażuje nadmierne zasoby nie zapewniając natychmiastowego usunięcia kontraktów; wydaje się jednak, że dokładność rzędu pojedynczych sekund jest do przyjęcia podczas niezbyt obciążających testów)<sup>2</sup>. W celu uproszczenia architektury prototypu, po dopuszczeniu i zestawieniu kontraktu program wsadowy sam uruchamia generatory ruchu na odpowiedniej relacji, symulując w ten sposób naturalne zachowanie użytkownika. W przypadku, gdy żądanie nie może być zrealizowane bezpośrednio po uruchomieniu programu, użytkownik oczekuje na zwolnienie zasobów (zakończenie innych kontraktów). Powyższy schemat odpowiada zakładanemu scenariuszowi funkcjonowania systemu w postaci docelowej.

Obecnie logika realizacji kontraktów wykorzystuje bezpośrednio sprzężenie zwrotne od stanu sieci (tj. od obciążenia łączy). Algorytm sprawdza dostępne pasmo na całej ścieżce od źródła do przeznaczenia – przy czym trasy ruchu są określone statycznie – i posługując się ustalonymi z góry współczynnikami mieszania ruchu decyduje o zestawieniu kontraktu tak, aby nie przekroczyć założonego stopnia multipleksacji ruchu (*overbooking*)<sup>3</sup>. Oszacowanie możliwości realizacji nowego kontraktu przebiega sekwencyjnie dla każdego urządzenia sieciowego na ścieżce danych, począwszy od routera abonenckiego. Zestawienie kontraktu jest zdeterminowane warunkiem:

$$c_{nRT} - \sum_i b_i^* - \gamma \sum_i b_i - \alpha b_{i+1} > 0 ,$$

---

<sup>2</sup> Ewidentne uproszczenie funkcjonowania logiki systemu uznano za dopuszczalne w implementacji testowej: aktywne przeglądanie listy kontraktów zużywa co prawda dodatkowe zasoby, nie wydaje się jednak by radykalnie wpływało na wyniki testów przy niedużym obciążeniu zgłoszeniami użytkowników.

<sup>3</sup> *Overbooking* oznacza możliwość dopuszczenia kontraktów, których sumaryczna deklarowana przepustowość jest większa od przydzielonego im pasma. Postępowanie takie jest możliwe oczywiście tylko w sytuacji, gdy zachodzi multipleksacja statystyczna, której warunkiem jest, by długoterminowa średnia przepustowość kontraktów pozostawała mniejsza od wartości zakontraktowanej (którą należy wtedy rozumieć jako potencjalnie dostępną wartość szczytową, czy średnią krótkoterminową). Takie podejście jest analogiczne do zasad organizowania ruchu w łączach tranzytowych w telekomunikacji z komutacją obwodów – można je stosować z powodzeniem w sieciach pakietowych dla olbrzymiej części ruchu obecnie obserwowanego w Internecie.

Tabela 1. Czasy rekonfiguracji urządzeń sieciowych podczas zakładania kontraktów

Przedmiot testu	Liczba kontraktów	Czas zakładania kontraktu [s]		Czas całkowity [s]
		pierwszego	ostatniego	
procedury komunikacyjne wersja 1	100	4	38	2221
wersja 2	100	3	6	352
program zarządzający	100	6	9	852

gdzie  $c_{nRT}$  – pasmo przeznaczone dla klasy nRT<sup>4</sup>,  $\sum_i b_i^*$  – pasmo rzeczywiście zajęte przez wszystkie realizowane kontrakty w klasie nRT (odczytane ze statystyki routera),  $\sum_i b_i$  – pasmo żądane łącznie przez wszystkie dotychczasowe kontrakty,  $b_{i+1}$  – pasmo żądane przez badany kontrakt. W powyższej nierówności łączne zakontraktowane pasmo ulega przeskalowaniu o współczynnik zapasu  $\gamma$ , a pasmo żądane przez badany kontrakt – przez współczynnik mieszania  $\alpha$ . Sens opisywanej heurystyki sprowadza się do zapewnienia kontraktowi pasma wynikającego z pewnego, ustalonego doświadczenia, współczynnika mieszania  $\alpha$  (zazwyczaj mniejszego od 1, co umożliwia *overbooking*), przy jednoczesnym zapewnieniu realizowanym kontraktom pewnego zapasu pasma związanego z ich „zrywnością”, tj. okresowym zapotrzebowaniem na pasmo większe, niż wynikające z obserwowanych statystyk (współczynnik  $\gamma$ ). W przypadku niemożności zrealizowania kontraktu program, korzystając z bazy danych kontraktów, oblicza czas, po którym powinno się zwolnić pasmo potrzebne do realizacji żądania, i wstrzymuje realizację żądania klienta aż do tego czasu.

## 5. Testy wydajnościowe

Przeprowadzono szereg testów weryfikujących poprawność proponowanej architektury systemu i szczegółowych rozwiązań technologicznych. Przetestowano system prototypowy ze względu na jego wydajność zarządzania kontraktami, ze względu na skuteczność priorytetyzacji ruchu w wykorzystywanej architekturze DiffServ, ze względu na stabilność pracy urządzeń sieciowych w sytuacji ich nieustannego rekonfigurowania oraz ze względu na skuteczność działania wybranego heurystycznego algorytmu dopuszczania połączeń. Te aspekty weryfikacji systemu zostały omówione w rozdziałach poniżej.

### 5.1 Wydajność zarządzania kontraktami

Celem testu było sprawdzenie wydajności zakładania kontraktów. Badanie przeprowadzono w dwóch etapach: w pierwszym sprawdzono wydajność biblioteki komunikacyjnej, w drugim – programu zarządzającego. W obu przypadkach routery nie były obciążone ruchem, co pozwala oszacować minimalny czas potrzebny na zestawienie kontraktu tak na komputerze sterującym, jak i na routerze. Testowano dwie wersje procedury zakładającej kontrakt, różniące się sposobem ustalania podziału pasma w klasie nRT. W *wersji pierwszej* przy zakładaniu nowego kontraktu modyfikowane są parametry wszystkich już istniejących kolejek DiffServ tak, aby uzyskać równomierny podział całego pasma w klasie nRT pomiędzy kontrakty. W *wersji drugiej* pasmo przydzielane w kontrakcie pozostaje w ustalonym stosunku do wartości średniej kontraktu. Obie wersje są równoważne w sensie podziału pasma między kontrakty (przy jednakowych żądanych wartości średnich w kontraktach), jednakże wersja druga charakteryzuje się mniejszym nakładem obliczeniowym wynikającym z uniknięcia pracochłonnej rekonfiguracji kolejek dla wszystkich założonych wcześniej kontraktów (należy wówczas odczytać parametry wszystkich kolejek, wyznaczyć nowe i rekonfigurować router).

<sup>4</sup> *Non real-time*, ruch elastyczny, poddający się mieszanemu statystycznemu, realizowany w klasie *assured forwarding* (AF) architektury DiffServ.

Wyniki testów zestawia Tabela 1. W trakcie pomiarów zaobserwowano pewne wahania czasu zestawienia kontraktu (rzędu 2-3 sekund), będące wynikiem zakłóceń transmisji i zmiennego obciążenia komputera wykonującego procedurę zarządzającą. Znacznie wolniejsze działanie procedur komunikacyjnych w wersji 1 jest wynikiem każdorazowego dopasowywania pasma przydzielonego poszczególnym kontraktom, co skutkuje zależnością czasu  $O(n^2)$  od liczby kontraktów  $n$ . Algorytm w wersji 2 pozwala osiągnąć akceptowalne czasy, należy jednak pamiętać, że może być konieczne okresowe przeliczenie i zmiana pasma wszystkich kontraktów.

Pomiar wydajności programu zarządzającego przeprowadzono tak samo jak w poprzednim teście. Wykorzystano procedury komunikacyjne w wersji 2 (tj. szybszej). Dodatkowy narzut wprowadzany przez program wynosi 2-5 s i rośnie z liczbą kontraktów. Wiąże się to z koniecznością analizowania istniejących kontraktów, co trwa proporcjonalnie do ich liczby.

**Wnioski.** Uzyskane czasy potrzebne do założenia kontraktu można określić jako akceptowalne, aczkolwiek uciążliwe dla kontraktów zawieranych na stosunkowo krótkie okresy czasu (rzędu minut). Niestety, procedury konfiguracyjne są wykonywane przez router sekwencyjnie, co ogranicza liczbę zakładanych kontraktów do kilku-kilkunastu na minutę<sup>5</sup>. Stanowi to poważne ograniczenie w masowym wykorzystaniu, jednakże w docelowym systemie możliwa jest pewna optymalizacja; przewiduje się, że również szybkość działania poleceń konfiguracyjnych w sprzęcie lepszej klasy niż testowany, jest odpowiednio większa.

## 5.2 Skuteczność zapewnienia jakości usług nRT w sieci

Testy w tej grupie miały na celu sprawdzenie technicznych możliwości zapewnienia jakości. Podano badaniu trzy zagadnienia:

- poprawność utrzymywania gwarancji w klasie nRT,
- możliwość współdzielenia łącza przez klasę nRT i *best effort* (BE),
- określenie maksymalnej liczby kontraktów możliwych w klasie nRT.

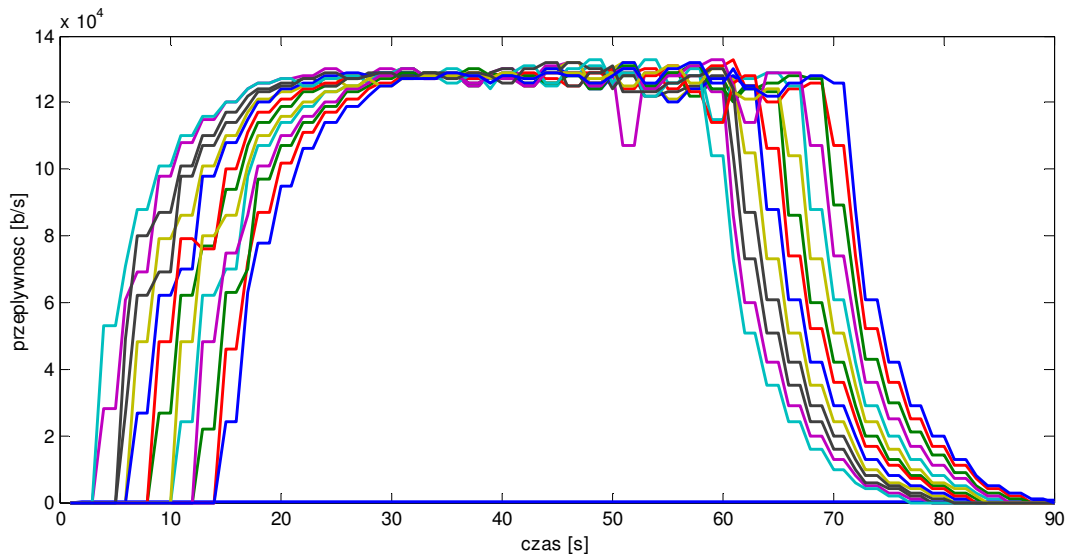
Klasa nRT przeznaczona jest głównie dla ruchu TCP i stosuje *shaping* w celu kształtowania parametrów ruchu – z tego powodu jako działanie poprawne należy określić sytuację, gdy otrzymane przez kontrakt pasmo jest zbliżone do zakontraktowanego i ustawionego jako średnia wartość filtru *shapingu*. Dopuszczalną rozbieżność należy przyjąć na poziomie pojedynczych procent (wynika ona z natury algorytmu TCP i ustawienia parametru *burst* – wydaje się, że korzystne jest pewne wygładzenie ruchu, skutkujące lepszym jego mieszanym w rdzeniu). Testy mające na celu określenie maksymalnej liczby kontraktów w klasie nRT, jaka może być obsłużona przez routery, mają na celu wykazanie możliwości powszechnego wprowadzenia kontraktowania pasma w rozległej sieci roboczej. Jednakże ze względu na możliwości sprzętu ich wynik należy traktować bardziej jako sprawdzenie szacunków niż ostateczną wartość.

Dla przebadania powyższych zagadnień przeprowadzono cztery eksperymenty mieszania rozmaitego ruchu w klasach nRT i BE:

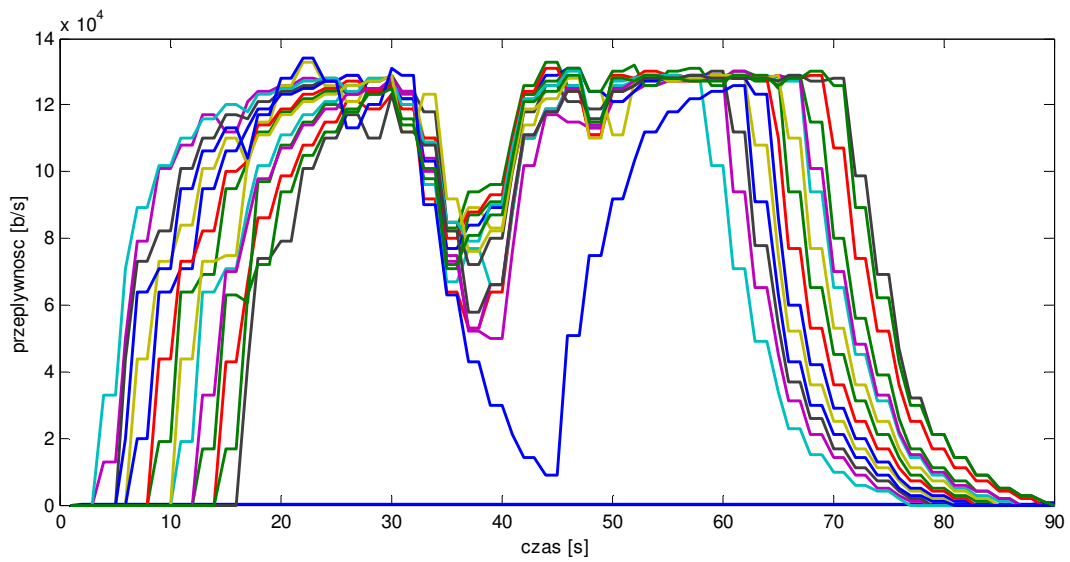
- a) 14 przepływów o zakontraktowanej wartości średniej 128 kb/s w klasie nRT,
- b) 15 przepływów o zakontraktowanej wartości średniej 128 kb/s w klasie nRT,
- c) 14 przepływów o zakontraktowanej wartości średniej 128 kb/s w klasie nRT oraz ruchu podkładowego w klasie BE,
- d) 30 przepływów o zakontraktowanej wartości średniej 64 kb/s w klasie nRT.

---

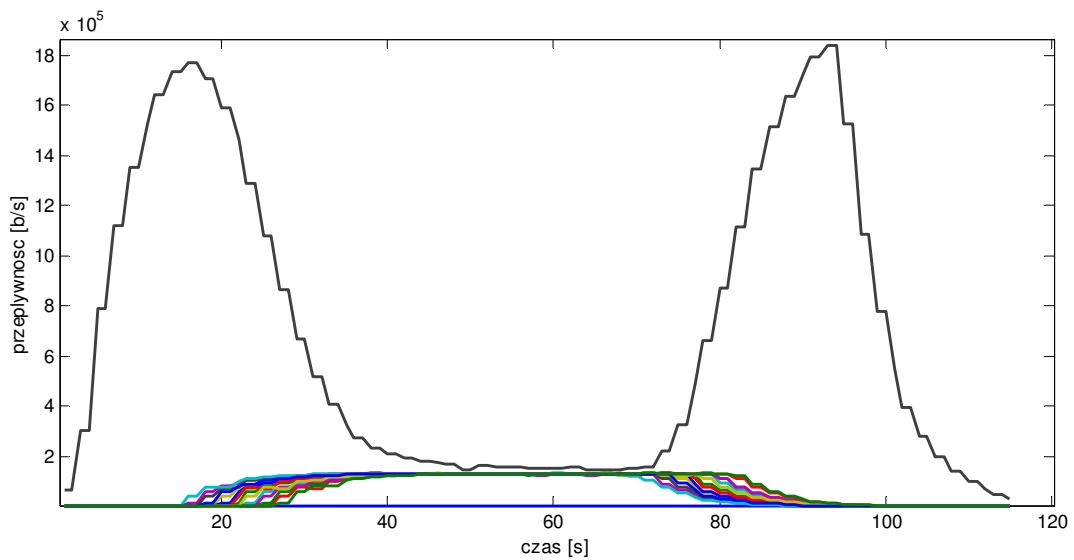
<sup>5</sup> Możliwe jest również uruchomienie i utrzymywanie kilku równoległych sesji zarządczych w routerze – wówczas oszczędza się czas potrzebny na połączenie, uwierzytelnienie i wprowadzenie routera w tryb konfiguracyjny. Niemniej jednak same polecenia konfiguracyjne wykonywane są przez system operacyjny sekwencyjnie. Kod opracowanej biblioteki komunikacji może więc stanowić przedmiot dalszej optymalizacji, aczkolwiek należy zdawać sobie sprawę z istniejących, zasadniczych ograniczeń dla przetwarzania równoległego poleceń.



**Rys. 3. Pasma uzyskane przez przepływy priorytetowe przy nieprzeciążonym łączu**



**Rys. 4. Pasma uzyskane przez przepływy priorytetowe przy łączu obciążonym w 93%**



**Rys. 5. Pasma uzyskane przez współbieżne przepływy priorytetowe i niepriorytetowe**

Należy dodać, że dla uniknięcia zakłóceń w procesie przesyłania danych kontrakty zostały założone przed uruchomieniem transmisji. Ruch w poszczególnych przepływach był generowany tak, by wypełniał całe dostępne pasmo. Przepustowość łącz szeregowych ustalono na 2 Mb/s. Wybór wariantów jest konsekwencją pojemności łącza – 14 kontraktów w eksperymencie a) zajmuje 1792 kb/s, czyli 87,5% dostępnego pasma, co oznacza, że łącze pozostaje nie w pełni obciążone. W eksperymencie b) wykorzystanie łącza wynosi 93%, czyli jest zbliżone do maksymalnego. Eksperyment c) przeprowadzono dla zbadania podziału pasma między klasę nRT i BE w warunkach silnego obciążenia routera, zaś eksperyment d) służy sprawdzeniu maksymalnej liczby obsługiwanych kontraktów.

Na zamieszczonych wykresach przedstawiono pasmo (średnie 30 sekundowe) uzyskiwane przez poszczególne przepływy (na osi rzędnych – pasmo w b/s, na odciętych – czas w s). Pomiarów dokonano na interfejsie routera dostępowego. Rys. 3 przedstawia **wyniki eksperymentu a)** – można zaobserwować, że w warunkach niepełnego obciążenia *shaping* działa poprawnie (dokładny pomiar wykazuje niezrównoważenie przydziału pasma rzędu 2%). Mimo występowania wahań wynikających z natury TCP, transmisję można określić jako stabilną, a podział pasma – sprawiedliwy.

Rys. 4 przedstawia **wyniki eksperymentu b)**, w warunkach większego (choć niepełnego) obciążenia. Sprawne działanie *shapingu* i zapewnienie zakładanych gwarancji jakości wynikających z proporcji podziału przepustowości łącza nie jest możliwe, co uwidacznia się załamaniem transmisji w środkowej części wykresu. Należy przypuszczać, że jest to wynikiem zmienności ruchu (wynikającej np. z synchronizacji algorytmów retransmisji TCP), skutkującej obciążeniem przekraczającym możliwości routera (w sensie łącz, procesora i/lub pamięci), gdyż obserwowany tutaj efekt nosi znamiona przepełnienia bufora, następującego po wystartowaniu wszystkich przepływów.

Rys. 5 przedstawia **wyniki eksperymentu c)**, gdzie współistniały klasy BE i nRT. Dla lepszego sprawdzenia, czy następuje wywłaszczanie, przepływ w klasie BE (odseparowana od innych linia na wykresie) został uruchomiony przed przepływami priorytetowymi. Eksperyment pokazuje, że podział pasma między klasy zachodzi w prawidłowy sposób, tzn. w okresie małego obciążenia możliwe jest pożyczanie pasma przez przepływ BE od klasy wyższej (nRT); natomiast ruch w klasie nRT wywłaszcza ruch w klasie BE. Średnie prędkości uzyskiwane przez przepływy w klasie nRT są zbliżone do tych osiąganych w wariancie A eksperymentu (również 14 kontraktów 128 kb/s, ale bez dodatkowego ruchu BE), co potwierdza prawidłowe działanie priorytetyzacji ruchu w kolejkach klasowych. Co więcej, osiągnięta sumaryczna przepływność (14 przepływów nRT po około 125 kb/s i BE – około 130 kb/s, czyli 1880 kb/s) jest większa niż w eksperymencie a), co świadczy wręcz o pożyteczności ruchu BE.

**Eksperyment d)** zakończył się niepowodzeniem – mimo pomyślnego skonfigurowania 30 kontraktów nie było możliwe przesłanie założonej liczby przepływów. Router przestał działać (nastąpiło przeładowanie systemu operacyjnego) po uruchomieniu 20. kontraktu, a więc w chwili, gdy obciążenie było dalekie od maksymalnego (tj. na poziomie ok. 63%) – co świadczy o obciążeniu, jakim jest dla routera *shaping*. Najbardziej prawdopodobnym powodem awarii był brak pamięci operacyjnej.

**Wnioski.** Kontrakty mogą być zestawiane prawidłowo pod warunkiem, że nie zajmują całego łącza a ich liczba jest ograniczona. Występowanie ruchu w klasie BE nie przeszkadza w realizacji kontraktów, może też stanowić znakomity “wypełniacz” dla pozostałego pasma. Realizacja kontraktu konsumuje znaczące zasoby routera, w szczególności routery serii 1700 pozwalają na utrzymywanie zaledwie kilkunastu kontraktów. Należy rozważyć typowe zastosowania systemu i wynikające z nich liczby kontraktów przypadające na router. Pozwoli to określić, do jakich użytkowników może być adresowana usługa elastycznych kontraktów, oszacować jaki sprzęt jest potrzebny i jakie są koszty i rentowność takiego przedsięwzięcia.

### 5.3 Zjawiska przejściowe przy zarządzaniu kontraktami i jakość usług RT

Testy w tej grupie miały na celu zbadanie możliwości współdzielenia łącza przez klasę RT (*real-time*) i BE. Poddano badaniu dwa ściśle związane ze sobą zagadnienia:

- skuteczności priorytetyzacji ruchu RT,
- efektów przejściowych występujących w fazie zakładania kontraktu i ich wpływu na jakość w istniejących przepływach RT.

Klasa RT przeznaczona jest do przesyłania ruchu o ścisłych wymaganiach czasowych. Z tego powodu konieczne jest utrzymanie niewielkich opóźnień pakietów, tak w sensie wartości średniej, jak i maksymalnej, oraz stosunkowo niedużej ich wariancji. W testach celem jest sprawdzenie możliwości wyłączenia klasy BE przez klasę RT przy zachowaniu przez tę ostatnią parametrów ruchu zbliżonych do sytuacji bez dodatkowego obciążenia. Transmisja w klasie RT jest realizowana w oparciu o protokół UDP. Brak sprzężenia zwrotnego w tym protokole pozwala, w odróżnieniu od TCP, wygodnie zmierzyć straty pakietów na łączach i wyrazić je w sposób bezwzględny.

Badanie efektów przejściowych oznacza sprawdzenie wpływu procesu przekonfigurowania routerów na istniejące przepływy RT – jest ono szczególnie istotne, gdyż tylko w sytuacji braku zakłóceń wywołanych przez taką operację możliwe jest realizowanie kontraktów dynamicznych RT. Zakłócanie klasy RT przez proces rekonfiguracji ogranicza, a raczej wyklucza użytkowanie systemu.

W eksperymentach badających **skuteczność priorytetyzacji RT** symulowano jednoczesną emisję pięciominutowych fragmentów filmu w formacie H.263, o średniej szybkości transmisji 256 kb/s, w rozdzielczości 176x144 piksele – a więc transmisji wideo typowej dla telefonii komórkowej. H.263 jest koderem typu CBR (*constant bitrate*), lecz zanotowane przepływności chwilowe osiągają w tym przypadku 1,4 Mb/s. Eksperymenty polegały na jednoczesnej emisji 1 lub 2 strumieni wideo w klasie EF (*expedited forwarding*) DiffServ oraz 4 lub 6 przepływów w klasie BE. W odróżnieniu od testów przedstawionych dotychczas, przepływy priorytetowe współdzieliły jedną kolejkę klasową, której przydzielono 90% dostępnego pasma. Tabela 2 przedstawia zaobserwowane wartości parametrów jakości transmisji wideo: zapisano w niej czcionką pogrubioną odsetek pakietów utraconych w poszczególnych transmisjach oraz, czcionką pochyłą, statystyki opóźnień odnotowane dla całego ruchu priorytetowego (opóźnienie minimalne/średnie/maksymalne [ms]).

Tabela 2. Stratność i opóźnienie dla transmisji priorytetowej UDP

		Klasa priorytetowa	
		1 przepływ	2 przepływy
Klasa BE	4 przepływy	<b>0</b> <i>20/39/85</i>	<b>1,3</b> oraz <b>7,8</b> <i>20/44/86</i>
	6 przepływów	<b>0</b> <i>20/42/85</i>	<b>6</b> oraz <b>0,5</b> <i>20/45/86</i>

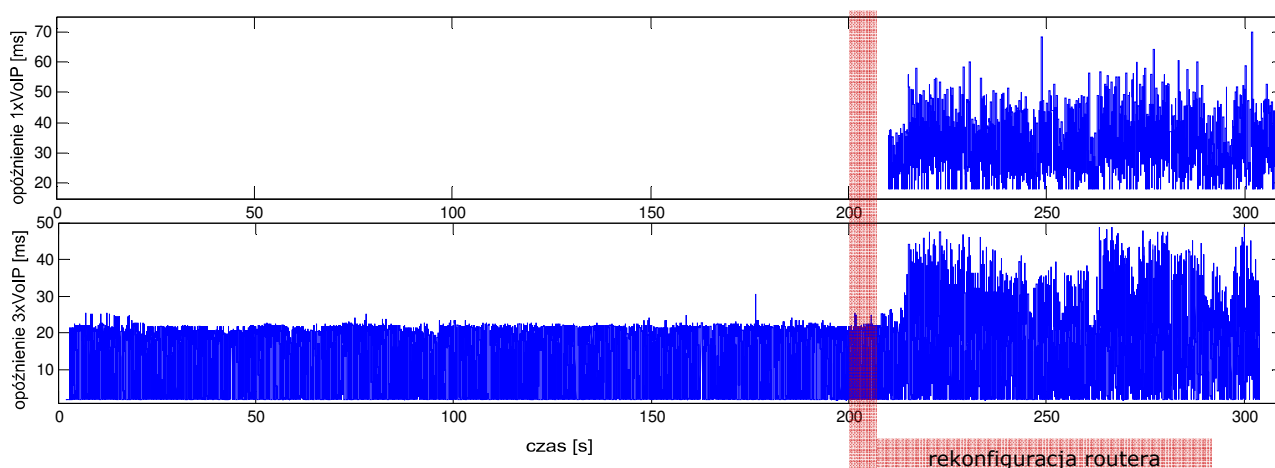
**Wnioski.** Testy udowodniły bardzo dobre funkcjonowanie mechanizmu kolejek priorytetowych DiffServ. Przy pojedynczym przepływie w klasie priorytetowej pakiety praktycznie nie są tracone, a obserwowany duży rozrzut opóźnień wskazuje na skuteczne buforowanie pakietów, co chroni przed ich utratą nawet przy tak dużej szybkości szczytowej. Opóźnienia z tytułu buforowania nie są jednocześnie na tyle duże, by uniemożliwiały prowadzenie wideorozmowy w sposób komfortowy. Użycie tej samej klasy priorytetowej do transmisji dwóch strumieni zamiast jednego skutkuje większym odsetkiem traconych pakietów, wynikającym z zastosowania stosunkowo niewielkich (rzędu 8 pakietów) kolejek w routerach. Krótkie kolejki gwarantują z kolei ograniczone, akceptowalne czasy opóźnień. Stopień ogólnego obciążenia routera, determinowany przez liczbę transmisji w klasie BE, nie wydaje się wpływać na skuteczność priorytetyzacji. Konkludując, można stwierdzić, że badane routery znakomicie nadają się do priorytetyzowania ruchu UDP.

W eksperymencie badającym **efekty przejściowe** dokonywane jest sprawdzenie, czy transmisje już zakontraktowane nie doznają uszczerbku na jakości w momencie rekonfigurowania routera przy



zakładaniu nowego kontraktu. Mamy tutaj na uwadze ewentualne utraty pakietów lub nadmierne opóźnienia, występujące *zanim* rozpocznie się transmisja w ramach nowego kontraktu, i będące skutkiem niewydolności routera wskutek wykonywania poleceń rekonfiguracyjnych.

W teście uruchomiono 6 przekazów wideo jak poprzednio, jako ruch podkładowy w klasie BE. W klasie priorytetowej trwają natomiast 3 jednoczesne transmisje VoIP, których staranna obserwacja daje informacje o stanach nieustalonych w momencie zakładania nowego kontraktu. Transmisje VoIP wykorzystują protokół G.723.1, gdzie w pojedynczym przepływie 24-bitowa ramka nadawana jest co 30 ms.



Rys. 6. Opóźnienia dla transmisji VoIP oraz dla nowotworzonej transmisji wideo

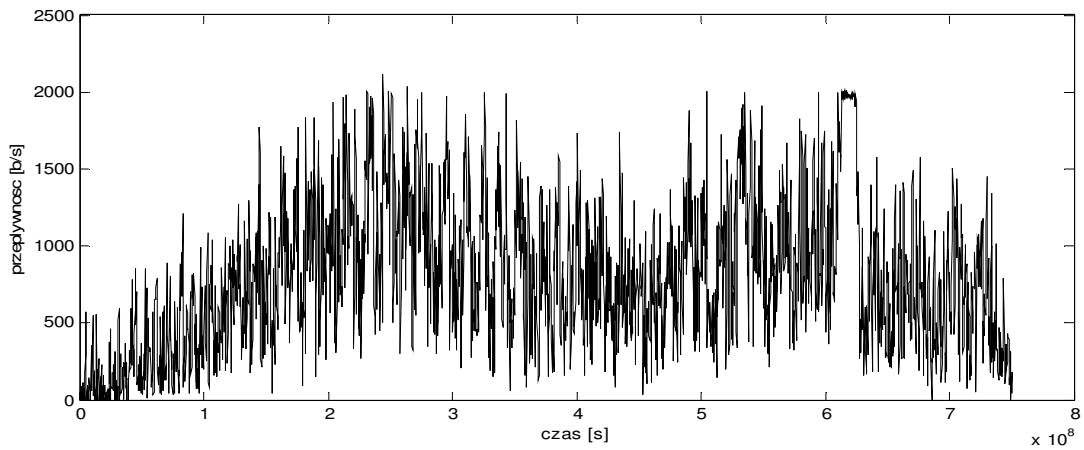
Rys. 6 przedstawia zaobserwowane opóźnienia dla połączeń VoIP (dolny wykres) i opóźnienia dla właśnie zestawianego kontraktu na ruchu VoIP (górny wykres) w funkcji czasu [s]. W scenariuszu testów, w dwusetnej sekundzie rozpoczyna się zakładanie kontraktu dla jeszcze jednego przepływu VoIP. Początek transmisji w ramach tego kontraktu ma miejsce w 210. sekundzie testu. Pod pojęciem „zakładania kontraktu” rozumiane jest stworzenie nowej pozycji w ACL (*access control list*), gdyż nowotworzony przepływ wideo „dołącza” do kolejki klasowej, z której korzystały dotychczas 3 przepływy VoIP.

**Wnioski.** Fakt zakładania kontraktu w żaden sposób nie znajduje odbicia w pogorszeniu jakości przekazów VoIP. Wynika stąd, że router może być przekonfigurowywany bez uszczerbku dla jakości realizowanej aktualnie transmisji – o ile sam router nie jest ewidentnie przeciążony.

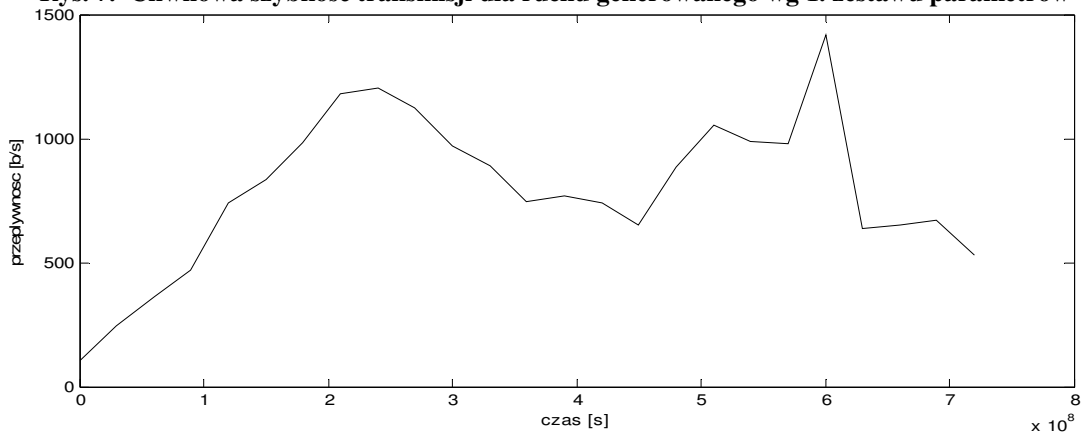
#### 5.4 Algorytm dopuszczania połączeń

Przedmiotem testów było sprawdzenie działania i dobranie parametrów strategii dopuszczania połączeń, realizowanej według algorytmu opisanego w rozdz. 4.3, której zadaniem jest maksymalizacja wykorzystania sieci (a dokładniej części pasma przeznaczonej dla klasy nRT), przy utrzymaniu gwarancji jakości dla kontraktów. Ponieważ założono, że użytkownicy klasy nRT używają transmisji TCP i są mniej wrażliwi na opóźnienie, podstawowym kryterium poprawnego działania algorytmu dopuszczania połączeń jest zapewnienie użytkownikom średniej przepustowości analogicznej jak przy korzystaniu z sieci nieobciążonej. Przykładowo taka transmisja towarzyszy przeglądaniu stron WWW, gdzie co pewien czas następuje przesył danych o różnej długości. Dobrym modelem takiego ruchu jest generator *on-off* z wykładniczym rozkładem czasu milczenia (*off*), a objętością transmitowanych danych modelowaną rozkładem Parety. Taki też model został użyty w eksperymentach. W celu sprawdzenia działania algorytmu dopuszczania połączeń w zależności od parametrów ruchu posłużono się dwoma zestawami parametrów generatora:

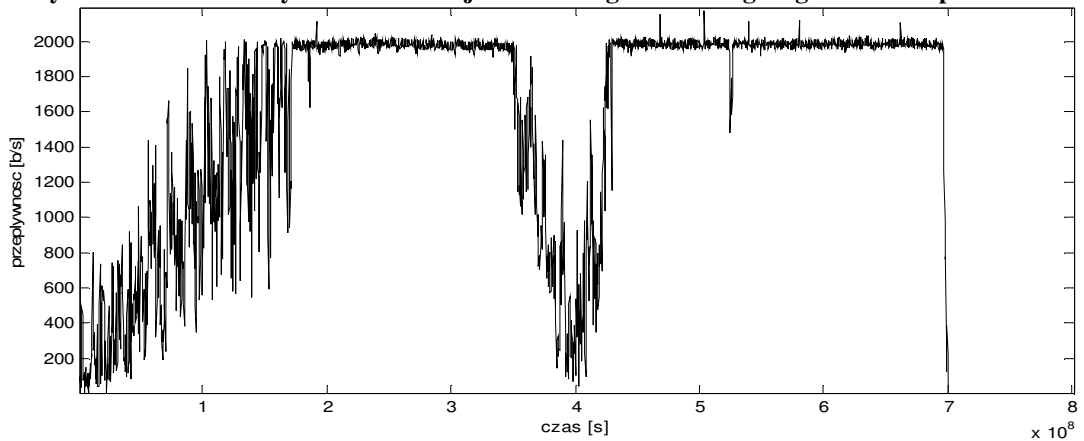
1.  $k=1,2$        $\sigma=512$  kb/s       $\lambda=2$     – średnia prędkość około 70 kb/s,
2.  $k=1,2$        $\sigma=512$  kb/s       $\lambda=5$     – średnia prędkość około 200 kb/s.



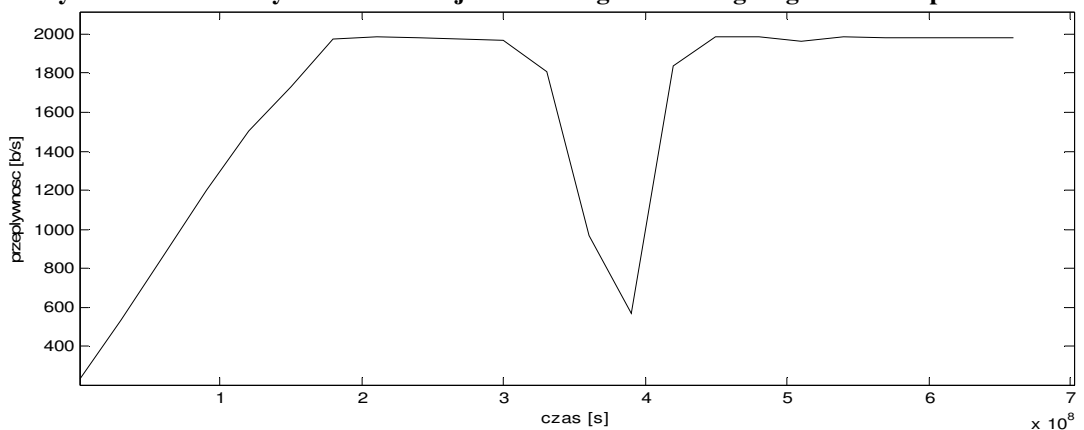
**Rys. 7. Chwilowa szybkość transmisji dla ruchu generowanego wg 1. zestawu parametrów**



**Rys. 8. Uśredniona szybkość transmisji dla ruchu generowanego wg 1. zestawu parametrów**



**Rys. 9. Chwilowa szybkość transmisji dla ruchu generowanego wg 2. zestawu parametrów**



**Rys. 10. Uśredniona szybkość transmisji dla ruchu generowanego wg 2. zestawu parametrów**

Parametr  $k$  oznacza współczynnik kształtu rozkładu Parety<sup>6</sup>,  $o$  – maksymalną dopuszczalną prędkość transmisji, a  $\lambda$  – częstość zdarzeń w rozkładzie wykładniczym (odwrotność średniego czasu między zdarzeniami). W obu wariantach co 5 s podejmowana jest próba założenia nowego, trwającego 300 s, kontraktu (jak już wspomniano, kontrakty niemożliwe do zrealizowania są odraczane). W sumie zgłaszanych jest 30 kontraktów, przez co w obu wariantach zapotrzebowanie znacząco przekracza dostępne pasmo. Żądane łączne pasmo kontraktów wynosi 15 Mb/s na 2 Mb/s dostępne, a rzeczywiste wartości średnie ruchu (ok. 2,1 Mb/s w pierwszym przypadku i 6 Mb/s w drugim) są również większe od dostępnego pasma. Do osiągnięcia właściwego działania sieci nie wystarczy więc sama multipleksacja ruchu, lecz przede wszystkim potrzebny jest sprawnie działający algorytm dopuszczania połączeń.

Rys. 7 przedstawia krótkoterminowe średnie prędkości transmisji (okres uśredniania 0,3 s) zebrane z interfejsu maszyny odbierającej (na osi rzędnych przedstawiono prędkość transmisji w kb/s, na odciętych – czas w s), dla pierwszego zestawu parametrów generatora. Rys. 8 przedstawia długoterminowe średnie prędkości transmisji (okres uśredniania 30 s), zebrane z interfejsu maszyny odbierającej (opis osi jw.), dla pierwszego zestawu parametrów generatora.

Oba wykresy przedstawiające przebieg eksperymentu dla pierwszego zestawu parametrów generatora wykazują spadek prędkości transmisji w środkowym okresie czasu. Wynika on z faktu, że prawie połowa kontraktów została odroczone o czas trwania kontraktów zawartych wcześniej (300 s), podczas gdy ruch był generowany przez 200s. Ruch charakteryzuje znaczna zrywność – średnie krótkoterminowe sugerują, że interfejs osiągnął – co prawda przez krótki okres – kres swych możliwości, natomiast średnie 30-sekundowe wskazują znaczne niedociągnięcie. Wydaje się, że współczynniki algorytmu dopuszczania połączeń są zbyt asekuracyjne, aczkolwiek w sensie ruchu deklarowanego osiągnięty współczynnik multipleksacji jest znaczny (15 jednocześnie aktywnych kontraktów daje  $15 \cdot 512 \text{ kb/s} = 7,68 \text{ Mb/s}$  na 2 Mb/s dostępne). Jednakże rzeczywiście (tzn. w sensie długoterminowych średnich) sieć pracuje w warunkach niedociągnięcia ( $15 \cdot 70 \text{ kb/s} = 1,05 \text{ Mb/s}$ ), a shapery nie wygładzają ruchu w zauważalny sposób.

Rys. 9 przedstawia krótkoterminowe średnie prędkości transmisji (okres uśredniania 0,3 s) zebrane z interfejsu maszyny odbierającej dla drugiego zestawu parametrów generatora, natomiast Rys. 10 – długoterminowe średnie prędkości transmisji (okres uśredniania 30 s) dla drugiego zestawu parametrów generatora. Dla tego wariantu parametrów, podobnie jak poprzednio, około połowa kontraktów została odroczone. Biorąc jednak pod uwagę rzeczywiste długoterminowe średnie prędkości generowanych przepływów, dopuszczono do zestawienia stanowczo zbyt dużej ich liczby naraz, dlatego doszło do przeciążenia łącza. Widać, że w tym przypadku niezmienną strategią dopuszczania połączeń okazała się stanowczo zbyt liberalna. Przy tak dużym przeciążeniu i tak niejednostajnym ruchu dochodzi nie tylko do ograniczenia pasma kontraktów, ale też do wyjątkowo niesprawiedliwego podziału pasma – osiągnane wartości prędkości transmisji przez poszczególne przepływy oscylowały między 30 a 150 kb/s.

**Wnioski.** Nie jest możliwe dopuszczanie kontraktów o znacznym zróżnicowaniu charakteru ruchu przy pomocy heurystyki posługującej się współczynnikami multipleksacji, które nie mogą precyzyjnie oddać zmienności transmisji. Czynnikiem utrudniającym precyzyjne dopuszczanie ruchu jest dodatkowo użycie do określenia stanu sieci średnich 30-sekundowych, odczytywanych z interfejsów routerów. Przy kontraktach zgłaszanych co 5 s średnie te nie mogą nadążać za rzeczywistym obciążeniem, co wprowadza znaczną niepewność (szczególnie niebezpieczną, gdy generowany ruch potrzebuje znacznego pasma, jak ma to miejsce drugim wariantem).

---

<sup>6</sup> Funkcja gęstości rozkładu Parety:  $f(x) = kx_m^k / x^{k+1}$ . Parametr  $x_m$  przyjęto w obu przypadkach równy 1200 B, co wydaje się rozsądną estymatą minimalnej ilości danych do przesłania w transmisji.

## 6. Podsumowanie

Przeprowadzone testy miały na celu weryfikację fundamentalnych założeń funkcjonowania projektowanego systemu dynamicznej kontraktacji usług przesyłowych z gwarancjami jakości:

- skuteczności priorytetyzacji przepływów w technologii DiffServ, w warunkach silnego obciążenia routera, zarówno poleceniami konfiguracyjnymi, jak i obsługiwanym ruchem;
- zdolności sprzętu sieciowego do częstego zakładania i usuwania kontraktów;
- przydatności zaproponowanego algorytmu heurystycznego dopuszczania połączeń, ze sprzężeniem od stanu sieci.

W opinii autorów, wyniki testów są optymistyczne i skłaniają do kontynuacji prac. Testy pokazały, że nawet skromne urządzenia sieciowe są w stanie wydajnie i stabilnie realizować rozmaite, dość mocno obciążające je kontrakty, w pełni wywiązując się z dyscyplinowania, kształtowania i monitorowania ruchu. Liczba uzyskanych jednocześnie kontraktów zupełnie wystarcza do zaspokojenia potrzeb gospodarstwa domowego.

Testy wskazały również słabe strony oprogramowania. Istnieją jednakże sposoby, by poprawić jego działanie. Wydajność komunikacji i rekonfigurowania routerów można nieco podnieść stosując przetwarzanie równoległe, natomiast weryfikację poprawności wydawanych poleceń można osiągnąć rozbudowując algorytmy analizujące konfigurację i statystyki routerów. W odniesieniu zaś do nie w pełni zadowalającego funkcjonującego algorytmu dopuszczania połączeń, można zastosować następujące środki zaradcze:

- estymację pasma zajętego w czasie ostatnich 30 s z wykorzystaniem bazy kontraktów – można w ten sposób uaktualnić wartość odczytaną z routera (opóźnioną o 30 s);
- powiązanie parametrów ruchu z konkretnym kontraktem, co umożliwiłoby stosowanie zawsze właściwych współczynników mieszania i zapasu pasma;
- stworzenie większej liczby klas o dobrze określonych własnościach ruchu tak, aby ograniczyć zmienność parametrów ruchu wewnątrz klasy.

Podsumowując, idea stworzenia kompletnego systemu rezerwacji i wyceny kontraktów przesyłowych nie jest trudna do urzeczywistnienia ani ze względu na aktualne możliwości technologiczne, ani ze względu na nakład pracy, ani ze względu na niekorzystne uwarunkowania strukturalne w sieci (brak skalowalności lub odpowiednich modeli ruchu do wykorzystania). System taki w sposób ewidentny wspomagałby zarówno użytkowników, jak i operatorów. Niestety, rynek obecnie wydaje się niedojrzały do jego przyjęcia.

## Bibliografia

- [1] Castro R., Coates M., Liang G., Nowak R., Yu B. *Network Tomography: Recent Developments*, Statistical Science 19:3, str. 499-517, 2004.
- [2] Courcoubetis C., Weber, R. *Pricing Communication Networks: Economics, Technology and Modelling*, John Wiley & Sons, 2003.
- [3] *IP Solution Center – MPLS VPN*, white paper, Cisco Inc., 2003.
- [4] Kamola M., Arabas P. *Sieć z giełdowym systemem zawierania kontraktów na usługi przesyłowe – propozycja architektury*, Przegląd Telekomunikacyjny 8-9, str. 1142-1154, 2008.
- [5] Kelly F. *Notes on effective bandwidths*, Stochastic Networks: Theory and Applications (red. F.P. Kelly, S. Zachary, I.B. Ziedins), str. 141-168, Oxford University Press, 1996.
- [6] Lilien G.L., Kotler P., Moorthy K.S. *Marketing Models*, Prentice-Hall International Editions, 1992.
- [7] MERKATO, <http://www.invisiblehand.net>.
- [8] NS2, <http://www.isi.edu/nsnam/ns>.
- [9] Terpiłowski J., Skrzeczkowski A. M., Zych S. *Podział kosztów szkieletu – założenia*, raport NASK, 2005.
- [10] Wang X., Schulzrinne H. RNAP: A Framework for Congestion-Based Pricing and Charging for Adaptive Multimedia Applications.