VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

VLIV VÝPADKU LINKY A POUŽITÍ ALTERNATIVNÍCH TRAS NA ZAJIŠTĚNÍ KVALITY SLUŽEB

EFFECT OF NETWORK LINK FAILURE AND ALTERNATIVE ROUTE USAGE ON QUALITY OF SERVICE ASSURANCE

DIPLOMOVÁ PRÁCE

MASTER´S THESIS

AUTOR PRÁCE Bc. PETR ATANASČEV
AUTHOR

VEDOUCÍ PRÁCE Ing. JIŘÍ HOŠEK
SUPERVISOR

BRNO 2009

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor

Telekomunikační a informační technika

Student:	Bc. Petr Atanasčev	***ID:***	89648
Ročník:	2	***Akademický rok:***	2008/2009

NÁZEV TÉMATU:

Vliv výpadku linky a použití alternativních tras na zajištění kvality služeb

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou řešení výpadku linky a využívání záložních tras v IP sítích. Nastudujte technologii DiffServ, která slouží pro zajištění kvality služeb v datových sítích. Rozeberte možnosti použití alternativních komunikačních tras v rámci DiffServ domény. Poté v simulačním prostředí Opnet Modeler vytvořte model DiffServ domény, v které simulujte výpadek některé z páteřních linek. Sledujte vliv tohoto výpadku a použití náhradní trasy na průběh hlavních síťových parametrů. Zaměřte se zejména na parametry související se zajištěním kvality služeb.

DOPORUČENÁ LITERATURA:

[1] JHA, S.: Engineering Internet QoS. London: Artech House Publishers, 2002, ISBN: 1580533418
[2] SZIGETI, T., HATTINGH, C.: End-to-end QoS network design. Indianapolis: Cisco Press, 2005, ISBN: 1-58705-176-1.
[3] WANG, Z.: Internet QoS: Architectures and Mechanisms for Quality of Service. San Francisco: Morgan Kaufmann, 2001, ISBN: 1-55860-608-4.

Termín zadání:	9.2.2009	***Termín odevzdání:***	26.5.2009

Vedoucí práce: Ing. Jiří Hošek

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

Cílem této práce bylo seznámit se s technologií rozlišovaných služeb DiffServ, problematikou výpadku linky a nalezení alternativní trasy v DiffServ doméně.

V práci byly popsány v teoretické rovině vlastnosti rozlišovaných služeb, způsob jejich značení a zacházení v DiffServ doméně. Byl také objasněn pojem DiffServ doména, její struktura a stavový směrovací protokol OSPF, který je v současné době hojně používaný.

Na základě získaných znalostí byl v simulačním prostředí Opnet Modeler vytvořen funkční model DiffServ domény, ve kterém byl simulován výpadek linky hlavní trasy a její následné obnovení. Provedené simulace byly následně podrobeny analýze.

KLÍČOVÁ SLOVA

DiffServ, rozlišované služby, kvalita služeb, QoS, OSPF, Opnet Modeler

ABSTRACT

Information about the differentiated services (DiffServ), the link failure problems and the alternate routes in a DiffServ domain finding are the aims of the master´s thesis to be given.

Properties of the differentiated services, the usage of marking techniques in the DiffServ domain are described in the thesis. The concept of the DiffServ domain, its structure and mostly used today routing protocol OSPF are described too.

The function model of the DiffServ domain has been created in the simulation environment called Opnet Modeleler on the basis of the obtained knowledge. The link failure and the following link recovery have been simulated in the model and the effects have been analyzed after that.

KEYWORDS

DiffServ, differentiated services, quality of services, QoS, OSPF, Opnet Modeler

BIBLIOGRAFICKÁ CITACE

ATANASČEV, P.: *Vliv výpadku linky a použití alternativních tras na zajištění kvality služeb.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 74 s. Vedoucí diplomové práce Ing. Jiří Hošek.

PROHLÁŠENÍ

Prohlašuji, že svoji diplomovou práci na téma Vliv výpadku linky a použití alternativních tras na zajištění QoS v rámci DiffServ domény jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

podpis autora

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Ing. Jiřímu Hoškovi za velmi užitečnou metodickou pomoc a cenné rady při zpracování diplomové práce.

V Brně dne ....

podpis autora

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

ÚVOD

Komunikační standardy jsou již delší dobu formovány snahou o vytváření sítí s integrovanými službami, které jsou schopny nad jedinou infrastrukturou přenášet data, hlas nebo video. Tyto aplikace kladou na síť odlišné požadavky. Datový přenos je charakteristický proměnnými nároky na šířku pásma a na spolehlivost spojení. Hlas a video oproti tomu vyžadují relativně konstantní pásmo a garantovanou dobu doručení, přičemž částečná ztráta informace do určité míry není důležitá a lze ji kompenzovat různými opravnými metodami.

Aby síť efektivně využívala síťové prostředky z hlediska real-timeových aplikací, musí datům z těchto aplikací poskytovat určitou kvalitu služeb. Pojem kvalita služeb (Quality of Services - QoS) ve své podstatě znamená, že daná komunikační síť umí rozlišovat jednotlivé typy datového provozu a zacházet s nimi tak, aby splnila jejich požadavky na šířku pásma, zpoždění, jitter a ztrátovost paketů. V dnešní době je termín kvalita služeb především skloňován v souvislosti s dnešním Internetem. Internet jako celosvětová síť je založena na protokolové sadě TCP/IP, která sama o sobě není schopna kvalitu služeb zajistit. Jinými slovy to znamená, že není schopna rozlišovat jednotlivé druhy služeb např. real-time datové proudy od klasických datových přenosů a zachází s nimi zcela shodně [6].

Pro zajištění kvality služeb v počítačových sítích jsou v dnešní době na úrovni síťové vrstvy používány tři základní techniky, kterými jsou technologie „best effort“ (prakticky bez možnosti rozlišovat služby), technologie přepojování na základě značek (MPLS) a technologie rozlišovaných – diferencovaných – služeb (DiffServ). Cílem této práce je se v teoretické rovině seznámit s principem funkčnosti posledně zmiňované technologie, seznámit se s technikami používání alternativních tras a v praktické rovině se seznámit s vývojovým prostředím Opnet Modeler, ve kterém bude vytvořena modelová počítačová síť, do které bude následně implementováno QoS. Poté, co bude vytvořen funkční model sítě s alternativní trasou v DiffServ doméně, bude do sítě implementována vhodná technika pro nalezení alternativní trasy. Po vytvoření takového modelu sítě, budou provedeny simulace a jejich výsledky budou analyzovány.

1 TEORETICKÁ ČÁST

1.1 Seznámení se s QoS a jeho parametry

Kvalita služeb je nejen v počítačových sítích, ale i dalších spojově-orientovaných telekomunikačních sítích založena na rezervaci síťových prostředků, která předchází samotnému zajištění kvality dané služby.

Kvalita služeb je definována jako schopnost zajistit různým aplikacím, uživatelům, datovým tokům atp. různé priority. Například pro přenos dat není až tak důležité zajistit konstantní a co nejvyšší rychlost, ale spíše zajistit spolehlivost přenosu tak, aby nemohlo dojít ke ztrátě dat. Naopak u VoIP telefonie či videokonference požadavek na spolehlivost služby (přenosu) není tak důležitý jako požadavek na rychlost, jelikož ztráta dat je v tomto případě tolerována a data mohou být částečně rekonstruována pomocí rekonstrukčních technik [6].

Z těchto důvodů byly definovány čtyři základní parametry QoS, kterými jsou:

- šířka pásma
- zpoždění
- ztrátovost
- jitter

1.1.1 Šířka pásma

Při přenosu informací je jedním z rozhodujících aspektů objem dat, který je používaný přenosový kanál schopen přenést za určitý čas. Obvykle se v této souvislosti mluví (spíše neformálně) o přenosové kapacitě či propustnosti přenosové cesty. Správným měřítkem je však pouze přenosová rychlost (v bitech za sekundu)[6], [11].

Dosažitelná přenosová rychlost je ale vždy dána souhrnem fyzikálních vlastností přenosového média (vodičů, kabelů apod.) a vlastnostmi dalších technických prostředků, které přenosový kanál spoluvytvářejí (např. modemů, multiplexorů apod.).

Každý přenosový kanál je vždy schopen přenášet jen signály o frekvenci z určitého omezeného intervalu. Přesněji, signály s jinou frekvencí přenáší tak špatně (s tak velkým útlumem, zkreslením apod.), že není únosné jej pro přenos těchto signálů vůbec používat [11].

Šířka pásma je především kritická u multimediálních aplikací, ale někdy třeba i u nejběžnějšího přenosu WWW stránek.

1.1.2 Zpoždění

Celkové zpoždění nebo také jednocestné zpoždění je čas, který zabere datům cesta z vysílacího koncového zařízení do cílového koncového zařízení. Většina lidí u hlasových přenosů zpoždění zaregistruje, jakmile jeho hodnota převýší 150 ms. Jestliže převyšuje 200ms, je již kvalita přenášeného hlasu velmi špatná. Hodnota jednocestného zpoždění se skládá z těchto čtyř částí [6]:

- Propagační zpoždění - čas potřebný k cestě dat z jednoho konce sítě na druhý. Je způsobeno konečnou rychlostí šíření signálu po přenosovém médiu.
- Paketižační zpoždění je čas potřebný pro převod analogového signálu do digitálního a následně do rámců a jeho zpětný převod do analogového.
- Jitter-buffer zpoždení je zpoždění způsobené přijímačem uložením jednoho nebo více datagramů tak, aby výsledné zpoždění bylo konstantní.

Z hlediska velikosti zpoždění bývá zpravidla výhodné volit malé velikosti fragmentů. To ale způsobuje větší zatížení sítě, protože se zmenší rozdíl mezi velikostí hlavičky a daty, které paket obsahuje [6], [11].

1.1.3 Ztrátovost

Ztrátovost je definována jako poměr počtu odeslaných datagramů k počtu bezchybně přijatých datagramů. Datagramy, které jsou ztraceny a nemohou být obnoveny, vytvářejí u hlasových služeb a videokonferencí shluky chybějících datagramů. Pokud je ztráta datagramů rozložena náhodně, nevede to k tak významnému zhoršení kvality. Krátké shluky chybějících datagramů nevadí, ale vysoká ztrátovost datagramů, nebo ztráta většího množství datagramů následujících za sebou vede k výraznému zhoršení kvality. Naopak požadavky na nízkou ztrátovost jsou velmi vysoké u datových přenosů [15].

1.1.4 Jitter

Hodnota jitteru zachycuje velikost proměnlivosti příchozích časů datagramů od vysílače. Vysílací stana posílá datagramy v pravidelných intervalech. Ideálně by přijímací strana měla přijímat datagramy také v pravidelných časových okamžicích. V takovém případě by velikost jitteru byla nulová. Ale mnoho různých zařízení může určité datagramy v datové síti zpomalit, a tak některé datagramy přijdou dříve a jiné mnohem později. Jestliže

„pomalé“ datagramy jsou doručeny příliš pozdě, jsou pak vyřazeny, aby uvolnily místo datagramům, které následují za nimi [14].

1.2 Požadavky na DiffServ QoS

Požadavky na vysokou rychlost zpracování datových jednotek v uzlech sítě, procedury jako identifikace konkrétních sítových spojení, vyhodnocení aktuálních parametrů provozu, udržování stavových informací o jednotlivých spojeních atd., představují velice komplexní operace, které jsou náročné na výpočetní kapacitu [4].

Přitom v případě směrovačů je hlavní úlohou hledání nejvhodnější cesty pro doručení právě zpracovávaného paketu k cílovému uzlu a zajištění kvality služeb je pouze přídavnou funkcí. Z tohoto důvodu byla velká pozornost věnována mechanismům, které jsou schopny zajistit diferencované zpracování různých datových toků, ale přitom moc nezatíží procesor směrovače. To patřilo také mezi základní požadavky na technologii diferencovaných služeb.

Další analýzy ukázaly, že mezi časově nejvíce náročné operace patří identifikace datového toku, tedy třídění a kontrola dodržení předem sjednaných parametrů provozu, tedy měření. Správa front také představuje náročnou operaci, ale je třeba vzít v úvahu, že jsou fronty nedílnou součástí paketových sítí, a proto implementace nových obslužných metod představuje pouze rozšíření stávajících [14].

Aby bylo možné co nejvíce optimalizovat výkon potřebný k provozování systému pro zajištění kvality služeb, bylo třeba zredukovat místa v síti, na kterých jsou výkonnostně nejnáročnější operace prováděny. Místo prvního předpokladu, že každý směrovač bude zajišťovat ucelené zpracování paketů, u mechanismu DiffServ bylo zvoleno takové řešení, u kterého jednotlivé funkce byly rozděleny mezi sítové prvky. Tak byly definovány dva typy směrovačů - hraniční a páteřní, které společně zajišťují ucelený systém podpory kvality služeb [4].

1.2.1 Hraniční směrovače (Edge routery)

Rozdělení funkcí mechanismu diferencovaných služeb mezi jednotlivé směrovače ovlivňuje i vnitřní architekturu jednotlivých směrovačů. Hraniční směrovače se nachází na hranici sítě s podporou mechanismu DiffServ, a proto musí být schopny příchozí tok datových jednotek klasifikovat, kontrolovat dodržení sjednaných parametrů rychlosti a odpovídajícím způsobem značkovat, pozdržet či v krajním případě zahodit tyto pakety.

Vnitřní architektura takového směrovače proto musí obsahovat příslušné funkční bloky. Předpokládá se, že funkce klasifikace, měření, značkování a zpracování jsou implementovány přímo na vstupu směrovače a takto předzpracované pakety jsou pak předány do směrovacího podsystému. Umístění výpočetně náročnějších funkcí do hraničních směrovačů má také výhodu v tom, že vytížení hraničních směrovačů je zpravidla výrazně nižší, než vytížení páteřních prvků. Ani intenzita provozu přicházejícího na hraniční směrovač není tak velká, jako v páteřní síti, kde je provoz koncentrován [7].

1.2.2 Vnitřní směrovače (Core routery)

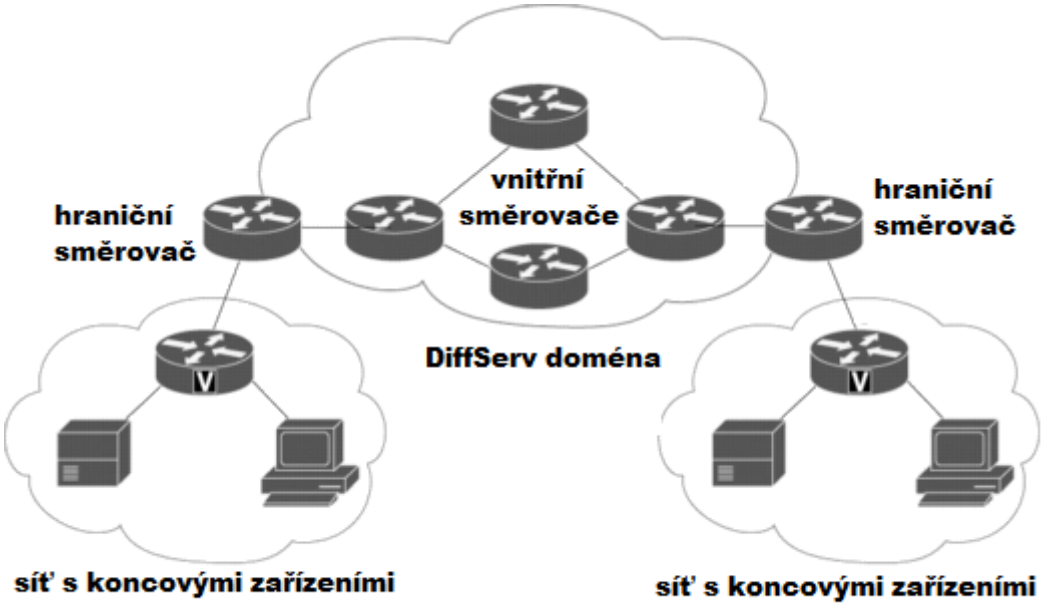
Jednotlivé hraniční směrovače jsou vzájemně pospojovány páteřními (vnitřními) směrovači. Z hlediska zajištění kvality služeb, páteřní směrovače už dostávají značkované pakety, kterým je již třeba pouze zajistit diferencované zacházení. Toto zacházení spočívá především v řízeném přidělování šířky pásma a zajištění specifikovaných parametrů zpoždění jednotlivým třídám [13].

Systém front bývá implementován na výstupu směrovače, což představuje ideální řešení z hlediska propustnosti celého systému. Protože funkce mechanismu DiffServ jsou úzce svázány se systémem front, i bloky tohoto mechanismu se musí nacházet v blízkosti těchto front. Modelová vnitřní architektura ukazuje, že pakety přicházející na odpovídající výstup musí být nejdříve klasifikovány na základě hodnoty DSCP, aby bylo možné je řadit do odpovídajících front. Fronty jsou pak obsluhovány plánovačem, který rozděluje šířku pásma mezi jednotlivé fronty a časovou předvídatelnost čekání paketu ve frontě podle nastavené konfigurace. Opět se jedná o modelovou architekturu, která v případě fyzické implementace může být odlišná v závislosti na fyzické realizaci spojovacího pole a systému front [13].

1.2.3 DiffServ doména

Část sítě složená z hraničních a páteřních směrovačů, které společně zajišťují určitou jednotnou sadu způsobů zacházení s pakety, je označena pojmem DiffServ doména. DiffServ doménu lze charakterizovat jako množinu vzájemně propojených síťových prvků, které zpracovávají datový provoz na základě stejných DiffServ pravidel. Kraje DiffServ domény musí být zajištěny hraničními směrovači. Ukázka struktury DiffServ domény je na obr. 1.1 [5].

Obecně lze rozlišit dva typy hraničních směrovačů. První typ se nachází na rozhraní mezi DiffServ doménou a sítí bez řízení kvality (využívající tzv. „best-effort“ způsobu přenosu). V tomto případě do hraničního směrovače přichází neoznačené pakety. Druhý typ hraničního směrovače se používá na rozhraní dvou DiffServ domén. V této situaci přichází k hraničnímu směrovači paket z jiné DiffServ domény, a proto lze očekávat, že již má přidělenu určitou značku. Určení značky paketu v druhé DiffServ doméně mají na starost pravidla implementovaná v jednotlivých DiffServ doménách. Pokud jsou tato pravidla stejná nebo velice podobná, je možné zachovat původní označení. V případě, že se používají podobná pravidla, ale jsou označována odlišnými identifikátory, může stačit náhrada starého identifikátoru novým. V uvedených dvou případech je ale nutné, aby způsoby zacházení odpovídající daným identifikátorům v jednotlivých DiffServ doménách byly smluvně definovány. V případě, že pravidla mezi DiffServ doménami jsou velmi rozdílná, je ignorována původní značka a musí proběhnout kompletní proces zpracování a přidělení identifikátoru [4].



Obr. 1.1: Příklad DiffServ domény

1.3 Princip DiffServ QoS

Obecně je princip QoS založen na klasifikaci datových jednotek spočívající v přidělení určité značky, která daná data třídí do předem definovaných tříd. V QoS je pak specifikován postup, jak zacházet s jednotlivými třídami, jak zajistit a měřit parametry síťového provozu a konečně postup řízení odesílání paketů, tedy typy front [13].

1.3.1 Třídění datových jednotek

Třídění datových jednotek je proces, kterým jsou pakety řazeny do skupin podle předem stanovených pravidel. Příkladem klasifikace (třídění) může být řazení paketů do front podle toho, ze které sítě přicházejí.

Proces klasifikace v síťových prvcích je prováděn na základě informací uložených v hlavičce datové jednotky. Dva nejpoužívanější typy třídění jsou: sloučené vyhodnocení (Behaviour Aggregate – BA) a vícepoložkové třídění (Multi-Field Classification – MF) [8].

BA třídění vybírá pakety podle jednoho jediného identifikátoru. Tímto identifikátorem je značka umístěná v záhlaví IP paketu v poli DSCP (viz. 1.3.4 – Differentiated Services Code Point). BA třídění se používá zejména v případech, kdy paket přicházející na směrovač, byl již označen dříve v jiném síťovém prvku a jeho třídění již tedy není třeba provádět [8].

Vícepoložkové třídění (MF) pak vybírá pakety na základě jedné nebo více položek v hlavičce protokolu IP, příp. TCP/UDP, jako jsou: zdrojová adresa, cílová adresa nebo typ, zdrojový/cílový port transportního protokolu či nějaká jejich kombinace [8].

Při přechodu datové jednotky z jedné části sítě do jiné může být značka zachována, nebo změněna na jinou značku se stejným významem - pokud různé DiffServ domény používají různé značky pro stejné zacházení - nebo změněna na jinou značku s jiným významem a to tehdy, pokud následující DiffServ doména nemůže zajistit zacházení s pakety použité v předcházející doméně. U paketů, které byly roztříděny v jiné části sítě, může hraniční směrovač tuto klasifikaci zachovat nebo změnit [16].

1.3.2 Značení paketů

Značkování slouží pro označení příslušnosti dané datové jednotky. Rozdíl mezi značkováním a tříděním spočívá v tom, že značka přidělená paketu může být využita ke třídění, zatímco třídění je proces zpracování již označené datové jednotky.

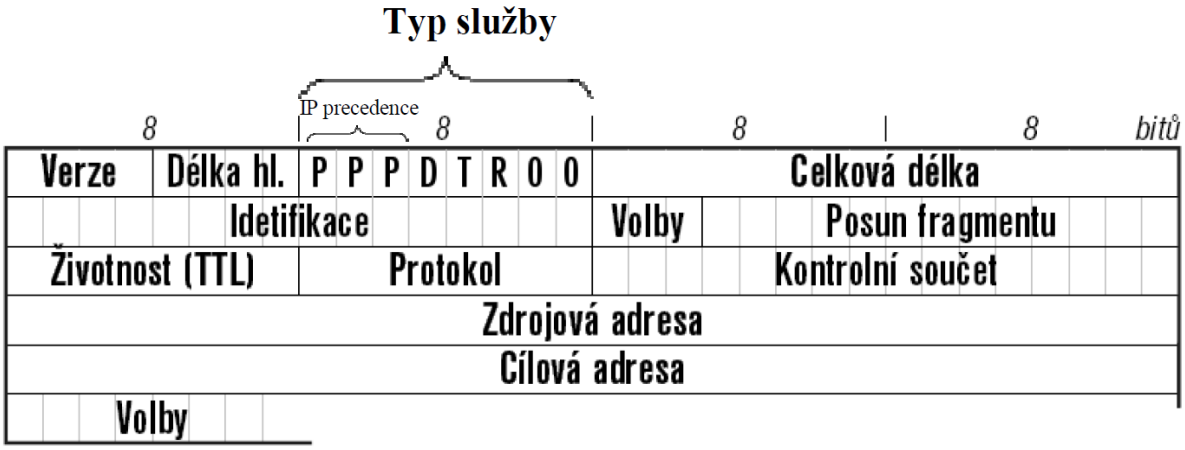
Značkování je obvykle realizováno nastavením hodnoty určitého pole v hlavičce IP datagramu. Příkladem takové značky může být IP adresa zdroje, IP adresa cílového uzlu, nebo jejich kombinace. Technologie DiffServ tedy nastavuje hodnotu pole DSCP (DiffServ Code Point) hlavičky IP pro identifikaci třídy.

Paket, který vstupuje do směrovače, již může být označen jiným prvkem sítě nebo zatím zůstat neoznačený. Jestliže byl paket již označkován, daný směrovač jej může přeznačit – třeba z důvodu vybočení paketu z předem sjednaných parametrů přenosu. Jiným důvodem přeznačkování může být případ, kdy paket přechází z jedné DiffServ domény do druhé DiffServ domény, kde se používají odlišná pravidla značkování [13].

1.3.3 Type of Service

Způsob značení paketu se odvíjí od technologie nebo protokolu užitého při přenosu paketu. Značka může být obsažena buď uvnitř hlavičky, nebo připojena k původnímu paketu. V hlavičce protokolu IP je obsaženo osmibitové pole, které se u IPv4 nazývá Type of Service (ToS). Jeho umístění v hlavičce IPv4 je vidět na obr. 1.2 [12].

ToS by měl zpravidla nastavit vysílač, aby určil charakter přepravní služby, která je pro daný diagram nejvhodnější. Pole se skládá ze tří bitů určujících prioritu dle specifikace typu služby, tří bitů, které určují požadavky na přenos a dvou nepoužívaných bitů, které v každém případě musí mít nastavenou hodnotu nula. Směrovač pak zpravidla při hledání cesty zohledňuje i požadavky definované nastavením pole ToS.



Obr. 1.2: Struktura protokolu IPv4

První tři bity pole ToS určují typ služby, tedy i absolutní prioritu při zpracování paketu ve směrovači. U těchto prvních třech bitů bylo nadefinováno osm úrovní, u kterých vyšší hodnoty znamenají vyšší prioritu. Konkrétně byly definovány následující prioritní úrovně [5]:

- routine (0) – rutinní práce, tj. zacházení typu „best-effort“,
- priority (1) – zprávy vyžadující zrychlené zpracování,
- immediate (2) – zprávy důležité z hlediska bezpečnosti státu či armády,
- flash (3) – varovné a další extrémně důležité zprávy, jako např. vojenské povely,
- flash Override (4) – zprávy rezervované pro využití v případě prolomení obrany či nukleárního útoku,
- CRITIC/ECP „Critical and Emergency Call Processing“ (5) – pro zprávy posílané autorizovanými zdroji ze státních složek civilní obrany,
- internetwork control (6) – pro zprávy určené pro řízení hraničních prvků sítě a spolupráce s jinými sítěmi,
- network control (7) – pro zprávy spojené s řízením sítě.

První tři bity ToS udávají pevně definovanou prioritu mezi jednotlivými kategoriemi provozu. Hodnoty 6 a 7 jsou určeny pro použití na řízení sítě.

Další bity pole ToS označují očekávanou metodu zpracování daného IP paketu:

- 3. bit (D) – indikuje požadavek na zpoždění (Delay) vybrané trasy:

    - 0 = normální zpoždění,
    - 1 = malé zpoždění,
- 4. bit (T) – indikuje požadavek na propustnost (Throughput) vybrané trasy:
    - 0 = normální propustnost,
    - 1 = vysoká propustnost,
- 5. bit (R) – indikuje požadavek na spolehlivost (Reliability) vybrané trasy:
    - 0 = normální spolehlivost přenosu,
    - 1 = vysoká spolehlivost přenosu.

Bity 6 a 7 jsou v současné době rezervované pro budoucí využití, avšak již dnes je vypracován návrh mechanismu pro řízení propustnosti, kde je počítáno s využitím těchto bitů [8].

Díky kombinaci šesti definovaných bitů a přidáním dvou dalších bitů se zatím nespecifikovanou funkcí vzniká značka, např. <10010100>, která označuje prioritu 4 (Flash Overdrive) a požadavek na malé zpoždění, normální propustnost a vysokou spolehlivost přenosu.

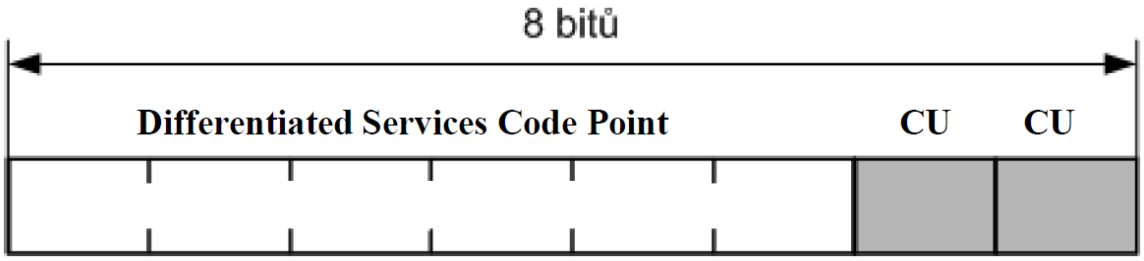
Pole ToS bylo původně určeno pro účely zpracování paketu ve směrovači, nebylo však běžně užíváno. Později začalo být toto pole využíváno pro podporu technologie DiffServ. U IP verze 6 jsou pak značky mechanismu DiffServ vkládány do pole Traffic Class (třída provozu). Strukturu protokolu IPv6 naznačuje obr. 1.3 [12].

Obr. 1.3: Struktura protokolu IPv6

1.3.4 Differentiated Service CodePoint (DSCP)

U diferencovaných služeb je zpracování provozu řízeno relativními prioritami přirazenými jednotlivým třídám provozu. Specifikace mechanismu Diffserv změní význam pole ToS a místo absolutní priority paketu toto pole udává pouze identifikátor třídy. Vzájemná priorita tříd je pak nastavena v rámci konfigurace daného směrovače.

Specifikace taktéž uvádí nové označení DS pro toto pole (viz. obr. 1.4). Dle této specifikace prvních šest bitů pole - tzv. kódové označení diferencované služby (DiffServ Code Point - DSCP) je používáno k označování tříd a zbylé dva bity opět zůstávají nevyužity a jsou označované jako „momentálně nevyužité“ (Currently Unusued – CU). Na základě DSCP je pak paketům zaručen žádaný způsob zacházení, neboli Per-Hop Behavior – PHB [13].



Obr. 1.4: Pole DS

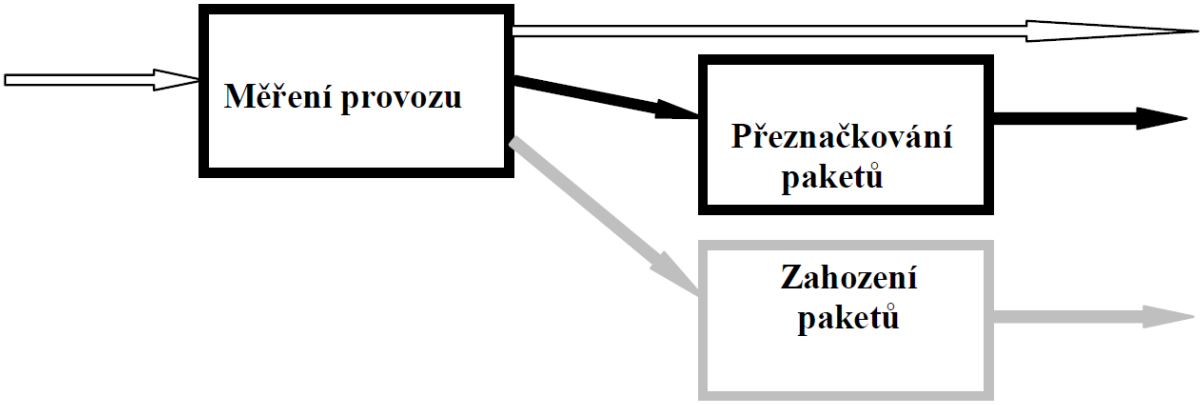
U DSCP se tedy používá zápis <XXXXXX00>. První bit označuje 0. bit, který má největší význam, oproti tomu poslední využitý bit je nejméně významný a označuje se jako 5. bit. Pomocí těchto bitů lze vyjádřit 64 různých hodnot, které jsou označovány výrazem „code point“ (CP).

Rozsah hodnot pole DSCP (6 bitů) byl rozdělen do tří skupin na základě účelu přidělování a následného řízení:

- blok 32 doporučených code pointů bylo standardizováno,
- blok dalších16 code pointů je vyhrazených pro pokusy nebo lokální využití,
- blok zbylých 16 code pointů je vyhrazen také pro pokusy a lokální užití, ale mohou být předmětem standardizace v případě, že dojde k vyčerpání hodnot z 1. kategorie.

1.3.5 Síťový provoz a jeho dohled

Dohled nad provozem by měl zajistit, aby datový tok vstupující do sítě se pohyboval v dohodnutých mezích. Dohled je složen ze dvou částí - měření provozu a dalšího způsobu zpracování paketu datového toku. Zvolený způsob zpracování může zachovat původně přidělené značky, přeznačit paket na jinou značku, či paket zahodit. Proto značkování odráží výsledek měření. Pokud se u označkovaného provozu během měření zjistí, že dochází k překročení dohodnutých parametrů, tak pak může dojít k přeznačkování, v rámci kterého je paketům přidělena značka s nižší prioritou. V případě potřeby pak síťové prvky mohou zpracování těchto paketů odložit, či dané pakety přednostně zahodit. Model bloku měření s následujícím zpracováním paketu je zobrazen na obr. 1.5 [8].



Obr. 1.5: Procesy dohledu nad síťovým provozem

Dohled nad síťovým provozem je založen na kontrole provozu přicházejícího na vstupní porty. Nejčastěji se ověřují následující dva parametry provozu:

Garantovaná průměrná přenosová rychlost (Committed Information Rate - CIR) - jedná se o typický parametr provozu, předem sjednaný mezi poskytovatelem připojení a uživatelem v dohodě o úrovni služeb (Service Level Agreement - SLA). Měří se v jednotkách byte/s a jejím výsledkem je rychlost přenosu IP paketů.

Maximální okamžitá přenosová rychlost (Peak Information Rate - PIR) specifikuje dlouhodobou průměrnou rychlost dat, jejichž přenos je zaručen uživateli v rámci dohody SLA. Dlouhodobá průměrná rychlost CIR je zpravidla menší než PIR [16].

Měření uvedených přenosových rychlostí vyžaduje sledování dalších parametrů, kterými jsou:

- velikost garantovaného shluku (Committed Burst Size - CBS),
- velikost maximálního shluku (Peak Burst Size - PBS),
- velikost nadměrného shluku (Excess Burst Size - EBS).

CBS definuje maximální velikost shluku dat, který může být zaslán maximální rychlostí PIR bez porušení dohody SLA. CBS tak specifikuje maximální počet paketů v bajtech, které mohou být přeneseny rychlostí PIR v jednom bloku. Velikost shluku EBS je prahová hodnota využívaná během měření objemu dat přesahující CBS. Pakety přesahující

EBS mají ještě menší prioritu, což většinou znamená jejich okamžité zahození. Parametr PBS má podobné využití jako CBS, ale je definován ve spojení s rychlostí PIR [8].

Při měření provozu je nejčastěji využívaným mechanismem mechanismus Token – Bucket (TB) založený na principu nádoby obsahující v každém okamžiku určitý počet tokenů. Výsledky tohoto měření jsou pak zohledněny při procesu značkování či rozhodování o zahození paketu.

Jakmile je identifikován datový tok, pro který byl sjednán předem definovaný způsob zacházení, proběhne měření, zda datový tok vyhovuje určeným parametrům. Na základě toho je pak paket označen příslušnou značkou. Celý tento proces se nazývá mechanismem barvení a v dnešní době jsou využívány dva základní mechanismy: jedna rychlost – tři barvy (srTCM) a dvě rychlosti – tři barvy (trTCM) [16].

1.3.6 Řízené odesílání paketů

Klíčem k zajištění odlišného zacházení různých datových toků ve směrovačích je řazení paketů do oddělených front a rozdílný způsob odesílání paketů z těchto front. Kromě samotného odesílání paketů podle odpovídajícího mechanismu, dalším důležitým úkolem řízení odesílání je dohled nad dostupnými sítovými prostředky, především nad šířkou pásma odchozího portu. Jelikož je technologie přepínání paketů založena na statistickém multiplexu paketu, není možné zaručit to, aby nedocházelo ke krátkodobému překročení kapacity odchozí linky. V takových případech jsou pakety s nižší prioritou pozdrženy ve frontách. Výběr paketů, které mohou být odeslány, řídí proces řízení odesílání. Mezi základní požadavky na metody řízení odesílání patří spravedlivé rozdělení dostupné šířky pásma mezi datové toky v souladu s prioritním systémem implementovaným do správy front a vyrovnaně mezi datové toky se stejnou prioritou.

U obecného směrovače pakety přicházejí na vstupní porty, které jsou předány spojovacímu poli. Na základě informací ve směrovací tabulce jsou tyto pakety přeneseny na požadovaný výstupní port. Každý výstupní port provádí klasifikaci paketu a řadí jej do patřičné fronty. Poté blok řízení určí, ze které fronty bude odeslán paket na výstup.

V dnešní době je používáno šest základních metod řízení odesílání paketů (front):

- fronta typu FIFO (First-In-First-Out),
- prioritní systém front (Priority Queuing - PQ),

- systém front se spravedlivou obsluhou (Fair-queuing – FQ),
- systém front s váženou cyklickou obsluhou (Weighted Round Robin - WRR),
- systém front s váženou spravedlivou obsluhou (Weighted Fair Queuing – WFQ),
- systém front založený na třídách s váženou spravedlivou obsluhou (Class-Based Weighted Fair Queuing – CB WFQ).

V DiffServ QoS se nejčastěji používá systém front založený na třídách s váženou spravedlivou obsluhou. Proto bude v této práci popsán pouze tento typ fronty.

1.3.6.1 Systém front založený na třídách s váženou spravedlivou obsluhou

U CB WFQ je příchozí provoz řazen do m tříd a šířka pásma odchozího portu je rozdělena do tříd podle váhové hodnoty přidělené těmto třídám, přičemž součet všech váhových hodnot je roven celkové šířce pásma, tedy

$$\sum_{i=1}^{m} w_i = 100\%, \tag{1}$$

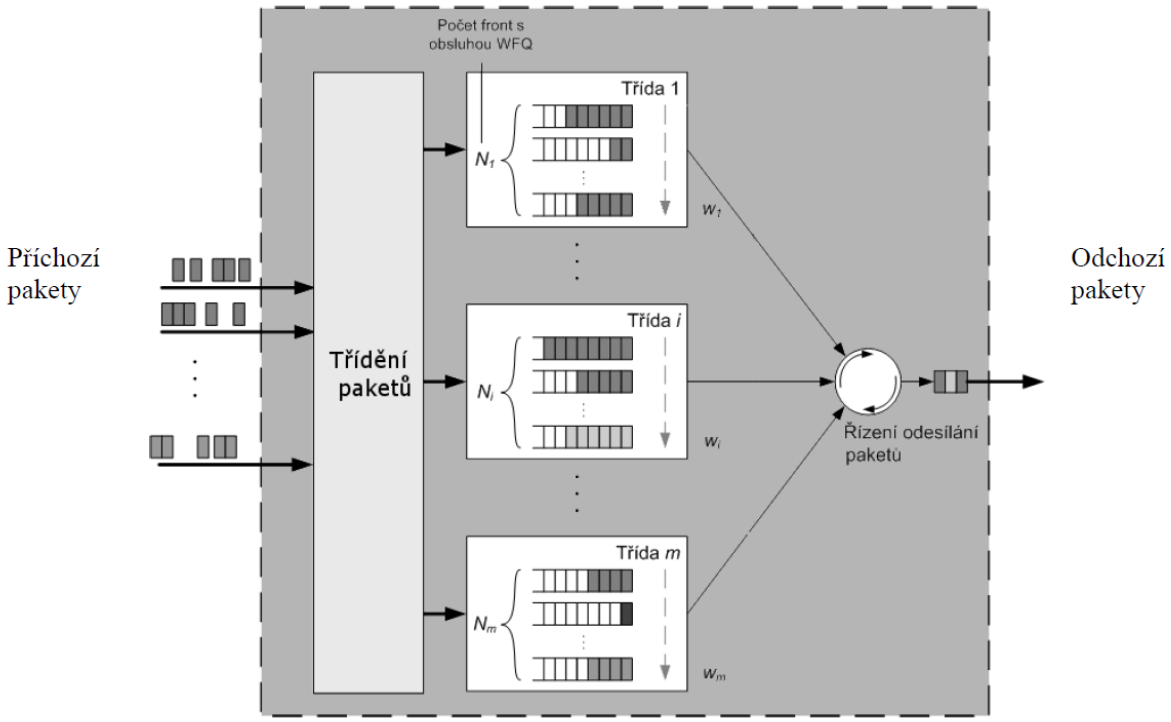
kde m je počet tříd a w_i je procentuální váha i-té třídy.

Pro obsluhu datových toků uvnitř tříd se využívá systém WFQ. Jestliže počet front obsluhovaných systémem WFQ v i-té třídě bude označen jako N_i, pak pro celkový počet front (s obsluhou WFQ) bude platit vztah

$$C = \sum_{i=1}^{m} N_i \,, \tag{2}$$

kde C je celkový počet WFQ front.

Pakety jsou odesílány tak, aby samotné odesílání skončilo v době vypočítané dle pomocného teoretického modelu zohledňujícího délku paketu. Řízení odesílání pak obsluhuje cyklicky každou frontu, ale jednotlivé pakety jsou na výstup posílány bit po bitu. Z tohoto důvodu musí větší paket čekat delší dobu, aby byl znovu složen. Blokové schéma systému front s váženou spravedlivou obsluhou řízenou podle tříd znázorňuje obr. 1.6 [5], [16].



Obr. 1.6: Fronta CB WFQ

1.4 Metody nalezení alternativních tras

V každé části sítě čas od času dochází k výpadku přenosové cesty, který je zpravidla způsoben přerušením fyzického spojení mezi prvky sítě, či výpadkem funkčnosti samotného aktivního prvku sítě. Proto je důležité sítě budovat tak, aby existovala vždy alternativní trasa, má-li být zaručena co možná nejvyšší spolehlivost sítě a kvalita služeb vůbec.

K tomuto účelu se využívají směrovací protokoly, jejichž primární funkcí je nalezení správné cílové adresy, která je umístěna mimo místní síť. Postupem času byly vytvořeny dva základní typy směrovacích protokolů, kterými jsou:

- protokoly využívající vzdálenost vektorů (distance vector)
- protokoly využívající stav síťových linek (link state)

Protokoly pro vzdálenost vektorů zakládají své rozhodnutí na vzdálenosti mezi zdrojem a cílem, měřené počtem směrování. Vliv u tohoto přístupu nemají argumenty typu velikost linek, aktuální zatížení, atp. Jejich princip spočívá v tom, že kratší cesta je ta lepší a rychlejší. Zpravidla používají Bellman-Fordův algoritmus, kdy si sousední směrovače

navzájem vyměňují své směrovací tabulky, ale topologii celé sítě neznají. Proto je také nevýhodné tyto směrovací protokoly nasazovat v rozsáhlých sítích. Do skupiny Distance Vector protokolů můžeme zařadit např. protokoly RIP nebo IGRP [1], [2].

Oproti tomu protokoly pro stav síťových linek rozhodují o nejlepší cestě podle několika aktuálních faktorů, včetně velikosti a aktuálního stavu linek, kterými musí daný paket projít. Vytváří tedy v paměti směrovače kompletní mapu celé sítě, označovanou jako topologická databáze (někdy se jí říká Link State Database). Nad touto databází potom pomocí algoritmu označovaného jako Shortest Path First (SPF) provádí výpočty potřebné k nalezení nejvýhodnější cesty do jednotlivých sítí. Mezi tzv. link state protokoly patří OSPF či IS-IS [1], [2].

Na základě výše zmíněných poznatků lze tedy konstatovat, že nejvhodnějším směrovacím protokolem pro nalezení alternativní trasy v DiffServ doméně se jeví směrovací protokol OSPF.

1.4.1 Směrovací protokol OSPF

Protokol OSPF (Open Shortest Path First) byl vytvořen organizací IETF přibližně v letech 1988 až 1991. Jeho nejnovější verze je definována v RFC2328. Tento protokol můžeme zařadit do skupiny směrovacích protokolů IGRP - Interior Gateway Routing Protocols. Je tedy určen k použití uvnitř jednoho autonomního systému (např. DiffServ domény).

Vychází z algoritmu SPF (Shortest Path Find), jehož prostřednictvím každý směrovač v síti vyhodnocuje nejlepší cestu paketu k libovolnému uzlu sítě. Prostřednictvím zpráv LSA (Link State Advertisement) provádí protokol OSPF kontrolu dostupných směrovačů a kontrolu stavu připojených linek. V zprávách LSA je popsán stav lokálního směrovače a linek vedoucích do sousedních směrovačů. Na základě těchto LSA zpráv jsou v každém směrovači sítě vytvářeny a aktualizovány tzv. databáze topologie sítě OSPF.

Z databáze sítě je v každém směrovači vytvořen stromový graf nejkratší cesty paketu v síti. Strom grafu reprezentuje aktuální hraniční směrovač dané oblasti, který poskytuje informace nejen o nejkratší cestě paketu, ale i o nejlepší alternativní cestě. Z těchto grafů jsou pak následně konstruovány vlastní směrovací tabulky OSPF. K aktualizacím prostřednictvím zpráv dochází v periodických časových intervalech. Nepřenáší se komplexní směrovací

tabulky, ale pouze změny ve stavu linek a dílčí změny, čímž se šetří přenosová kapacita linek [3].

1.4.1.1 Princip funkčnosti protokolu OSPF

Princip funkčnosti protokolu OSPF lze popsat pomocí následujících sedmi bodů [1]:

1. Směrovač vysílá přes svá rozhraní tzv. Hello pakety. Pokud se dva navzájem propojené routery pomocí těchto paketů dohodnou na určitých společných parametrech, stávají se sousedy (neighbors).
2. Mezi některými ze sousedů se vytvářejí užší vazby. Tyto routery se pak označují jako přilehlé (adjacent).
3. Přilehlé routery si vzájemně vyměňují pakety obsahující LSA (Link State Advertisement) informace. Ty popisují stav rozhraní směrovače nebo seznam směrovačů připojených k dané síti.
4. Všechny směrovače si ukládají přijaté LSA do své lokální topologické databáze a zároveň je přeposílají na ostatní přilehlé směrovače. Tím se informace postupně rozšíří mezi všechny směrovače v síti. Výsledkem bude shodná topologická databáze na všech směrovačích.
5. Po naplnění databáze každý směrovač provede výpočet pomocí SPF (Dijkstrova) algoritmu. Jeho výsledkem bude nalezení nejkratší cesty do každé známé sítě a odstranění smyček v topologii sítě.
6. Na základě vypočtených dat je možné naplnit směrovací tabulku routeru.
7. Pokud dojde ke změně topologie sítě, směrovač, na kterém ke změně došlo, odešle přilehlým směrovačům informaci v podobě LSA datových položek v OSPF paketu. Ta se postupně rozšíří po celé síti a každý směrovač upraví svou topologickou databázi a provede nový výpočet SPF algoritmu.

Velkou výhodou protokolu OSPF proti starším směrovacím protokolům (např. RIP) je jeho schopnost pracovat v relativně velkých sítích. Dosáhlo se toho zavedením dvou úrovní hierarchie. Síť je rozdělená na takzvané oblasti (area). LSA se běžně šíří pouze uvnitř dané oblasti a také výpočet SPF algoritmu se spouští pro každou oblast samostatně. Z jedné oblasti do druhé se předávají pouze sumární informace. Změna topologie sítě v jedné oblasti tedy nevyvolá přepočet SPF algoritmu v ostatních oblastech [1].

1.4.1.2 Výpočet ceny – metriky

Každý směrovací protokol potřebuje kritérium, podle kterého posoudí, která z více možných cest do cílové sítě je nejvýhodnější. Různé protokoly používají různá kritéria. Toto kritérium se označuje jako metrika.

Protokol OSPF používá metriku označovanou jako cena (cost). To je číslo v rozsahu 1 až 65535, přiřazené ke každému rozhraní směrovače. Čím menší číslo, tím má cesta lepší metriku a bude tedy více preferována. Standardně je ke každému rozhraní přiřazena cena automaticky odvozená z šířky pásma daného rozhraní podle vztahu (3).

$$cena = \frac{100.000.000}{sirka\_pasma}[bps] \tag{3}$$

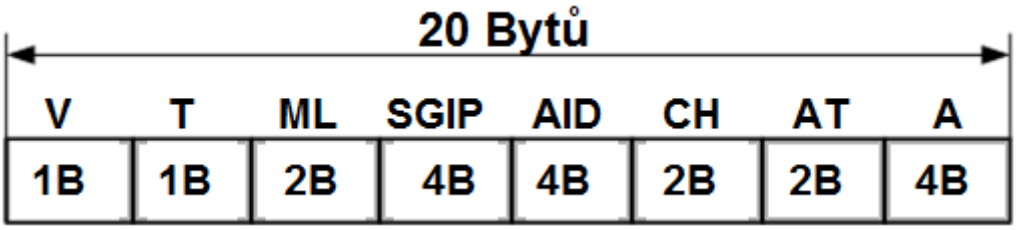
Například linka 64kbps bude mít standardně cenu 100 000 000/64 000 = 1 562. Aby tento automatický mechanismus fungoval, je třeba mít u každého rozhraní správně přiřazený bandwidth (šířku pásma). Dále je z uvedeného vztahu vidět, že linky FastEthernet a rychlejší budou mít shodně přiřazenou cenu 1. Proto některé implementace OSPF dovolují konstantu 100.000.000 změnit na vyšší. Dále je možné přiřadit cenu k rozhraní ručně a tím například upřednostnit pomalejší linku před linkou rychlejší.

Výsledná cena cesty ze směrovače do cílové sítě je pak dána součtem cen všech odchozích rozhraní, které se podílí na cestě paketu sítí [1].

1.4.1.3 Záhlaví protokolu OSPF

Záhlaví protokolu OSPF je zobrazeno na obr. 1.7 a je délky 20B. Protokol pro svou činnost používá následující typy zpráv [1], [10]:

- typ "HELLO" - dostupnost sousedních směrovačů
- typ "Databáze Description“ – určen pro budování databáze topologie sítě
- typ "Link State Request" – změna neaktuální části databáze
- typ "Link Status Update" – změna parametrů linek
- typ "Link State ACK" – potvrzení platnosti změny parametrů



Obr. 1.7: Záhlaví protokolu OSPF

Význam jednotlivých polí záhlaví protokolu OSPF znázorňuje následující tabulka (Tab.1.1) [1].

Tab. 1.1: Význam použitých symbolů záhlaví protokolu OSPF

Symbol	**Význam**
V (Version)	Verze protokolu OSPF
T (Type)	Typ zprávy (Hello, LSR, LSU, LSA,…)
ML (Message Length)	Délka zprávy
SGIP (Source Gateway IP)	Adresa vysílacího směrovače
AID (Area ID)	32bit dlouhý identifikátor oblasti
CH (Checksum)	Zabezpečení záhlaví
AT (Authentication Type)	Typ a způsob autorizace přístupu
A (Authentication)	Data používaná při autorizaci

2 PRAKTICKÁ ČÁST

Cílem tohoto projektu je vytvořit v prostředí Opnet Modeler model sítě se službami databáze VoIP, FTP a HTTP. Do takového modelu sítě dále implementovat zajištění kvality služeb, konkrétně služeb DiffServ. Po ověření správné funkčnosti vytvořeného modelu, bude implementován směrovací protokol OSPF zajišťující nalezení alternativní trasy při výpadku linky, který bude prováděn během simulací. Na základě těchto simulací bude porovnáván vliv výpadku linky na přenášená data a jejich kvalitu.

2.1 Úvod do Opnet Modeleru

Program OPNET Modeler (OM) je simulační prostředí, které bylo vyvinuto firmou OPNET Technologies Inc., a slouží pro návrh, simulaci a analýzu různých síťových technologií a mechanismů. Velice efektivně a podrobně dokáže modelovat chování rozsáhlých heterogenních sítí včetně komunikačních protokolů pracujících na různých úrovních modelu sítě [9].

2.2 Vytvoření základního modelu sítě

Základní model sítě je vytvořen pomocí editoru projektu. Prvky sítě jsou do projektu vloženy pomocí ikony palety objektů a jsou jimi klientské stanice (objekt ethernet_wkstn), servery (objekt ethernet_server), dva přepínače (ethernet16_switch) a čtyři routery (objekt ethernet4_slip8_gtwy), které tvoří výchozí model DiffServ domény – dva routery tvoří hraniční směrovače a dva vnitřní směrovače. Aktivní prvky sítě jsou navzájem propojeny ehthernetovou 10Mbit/s full-duplexní linkou (viz. obr. 2.1).

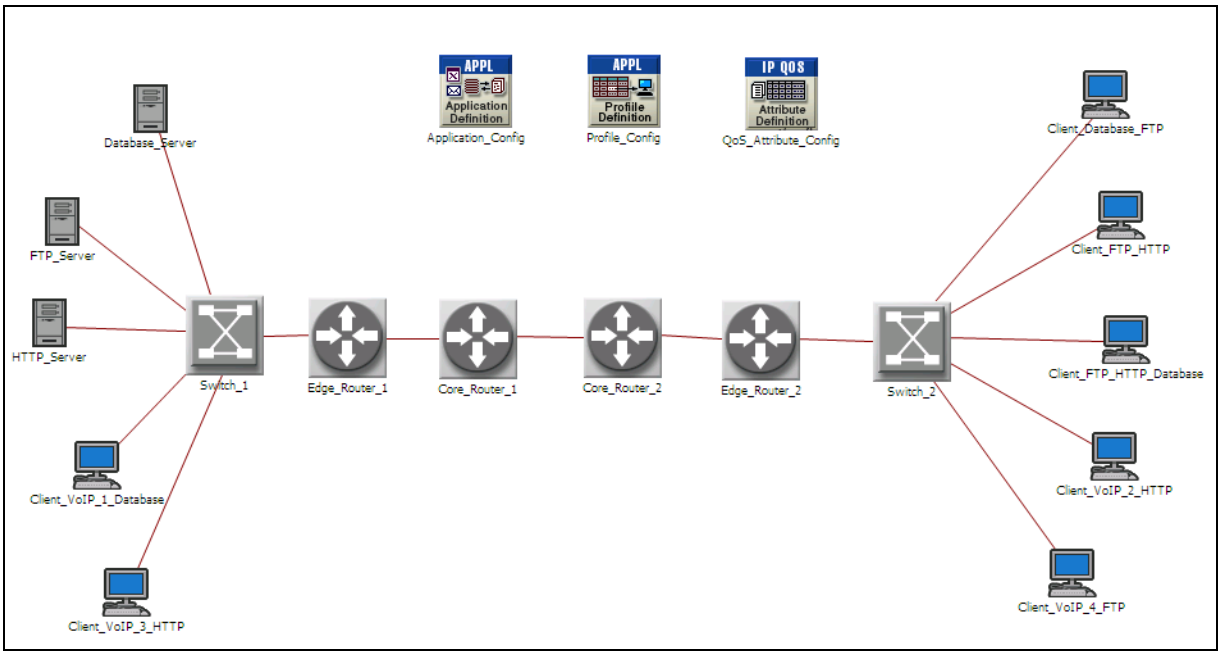
Součástí projektu jsou také tři objekty definující konfiguraci celého modelu sítě:

- objekt Application Config, ve kterém se nachází konfigurace aplikací použitých v modelu sítě
- objekt Profile Config sloužící k nastavení profilů jednotlivých aplikací, které budou následně volány koncovými prvky sítě
- objekt QoS Attribute Config, kterým jsou nastavovány atributy kvality služeb

Celý projekt se skládá z pěti scénářů. První scénář představuje funkční model sítě se všemi aplikacemi, druhý scénář je totožný s prvním – je však doplněn o zajištění kvality služeb a vytvoření základní DiffServ domény. Třetí scénář je totožný s druhým, pouze je

rozšířen o další vnitřní routery DiffServ domény tak, aby byla vytvořena alternativní trasa pro další simulace. Ve čtvrtém scénáři je do DiffServ domény implementován směrovací protokol OSPF, coby nejvhodnější směrovací protokol pro sledování stavu linky, s nastavením odpovídajících cen (metrik). V předchozích scénářích (scénářích 1, 2 a 3) je použito výchozího nastavení směrovačů dle Opnet Modeleru, takže na směrovačích je nakonfigurován směrovací protokol RIPv1. Pátý scénář je pak oproti čtvrtému scénáři rozšířen o objekt zajišťující výpadek linky v určitém čase simulace. Poslední scénář 6 pak kromě výpadku linky simuluje opětovné obnovení hlavní trasy DiffServ domény pro následnou analýzu vlivu tohoto obnovení na kvalitu služeb.

Další text se věnuje podrobnému popisu vytvoření jednotlivých scénářů.



Obr. 2.1: Schéma modelu sítě (scénář 1 a 2)

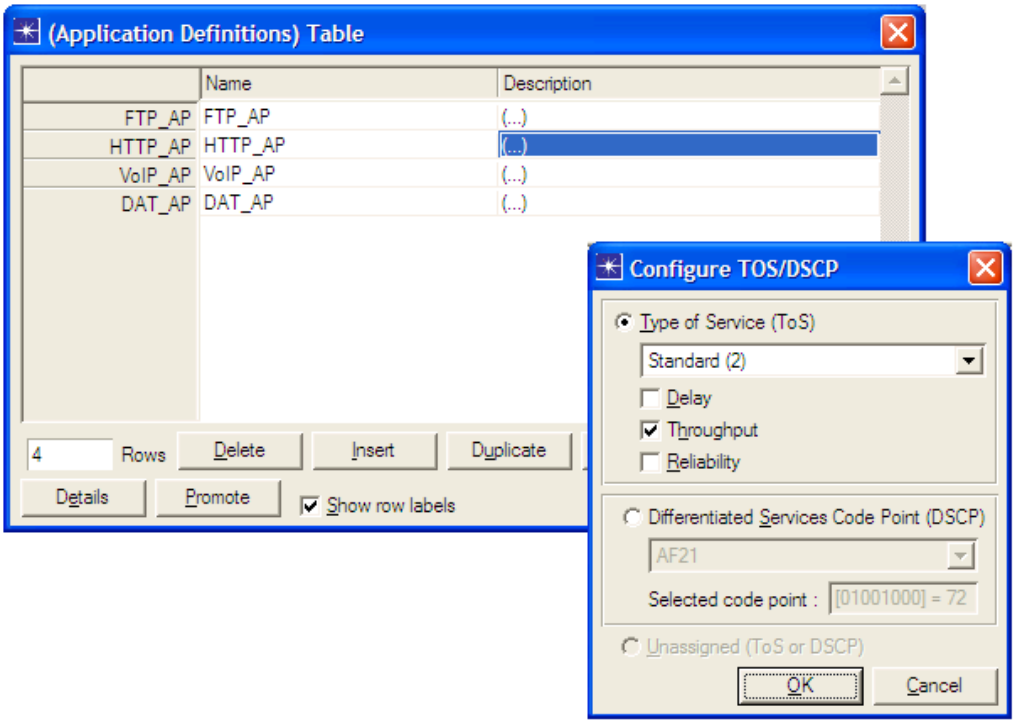
2.2.1 Scénář 1 – základní model sítě

Ve vytvořeném modelu je nejdříve nutné nakonfigurovat služby a profily, které budou využívat jednotlivá koncová zařízení.

Služby (aplikace) se, jak již bylo řečeno, nastavují v Application_Config, konkrétně v kontextové nabídce pod položkou Edit Attributes. V tabulce definice aplikací (Application Definitions) byly vytvořeny čtyři řádky, přičemž každému řádku odpovídá nastavení jedné

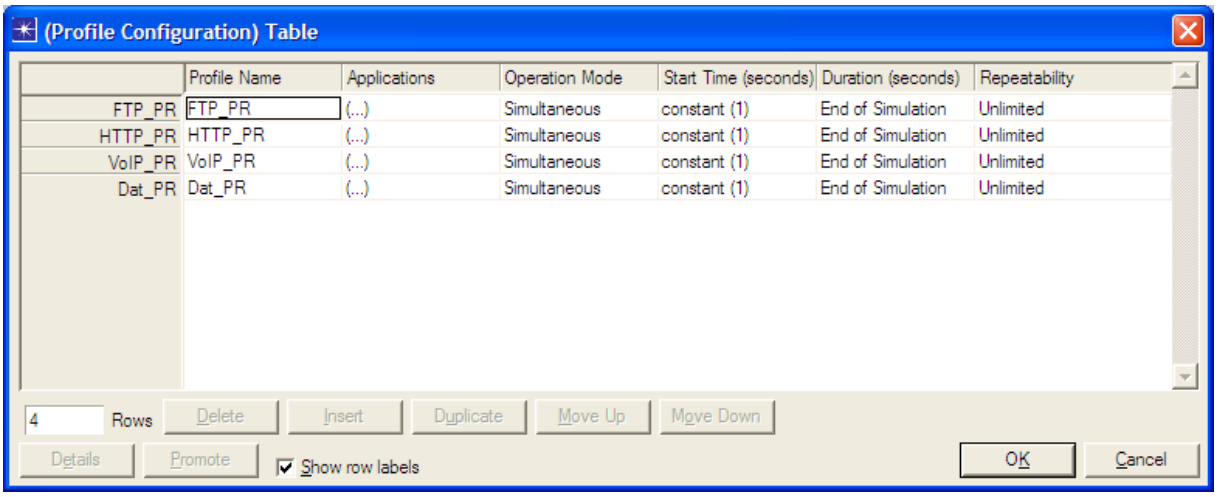
služby, tedy FTP, HTTP, VoIP a databáze (viz. obr. 2.2). Každé aplikaci je možné nastavit různé parametry. V případě tohoto modelu zůstaly všechny hodnoty ve výchozím nastavení kromě velikosti souboru (File Size) u FTP, vlastností stránek (Page Properties) u HTTP a množství komunikace s databázovým serverem (Transaction Size). Tyto hodnoty byly zvýšeny z důvodu většího zatížení sítě, aby následně bylo možné lépe vyhodnocovat výsledky naměřených simulací.

V nastavení profilů (Profile Config) je třeba vytvořit profily (viz. obr. 2.3). Jejich konfigurace probíhá obdobně jako u konfigurace služeb – v tabulce Profile Configuration se vytvoří opět čtyři řádky jednotlivých služeb a každému řádku se ve sloupci Applications přiřadí odpovídající, již vytvořená, aplikace. Ostatní položky tabulky zůstanou ve výchozím nastavení.



Obr. 2.2: Definice aplikací v Application_Config s ukázkou nastavení ToS

Nyní je třeba nastavit všechna koncová zařízení. Na následujících řádcích bude popsán postup pouze u jednoho ze serverů (konkrétně FTP) a jedné z klientských stanic (konkrétně stanice pojmenovaná Client_VoIP_2_HTTP), jelikož konfigurace ostatních zařízení je analogická.

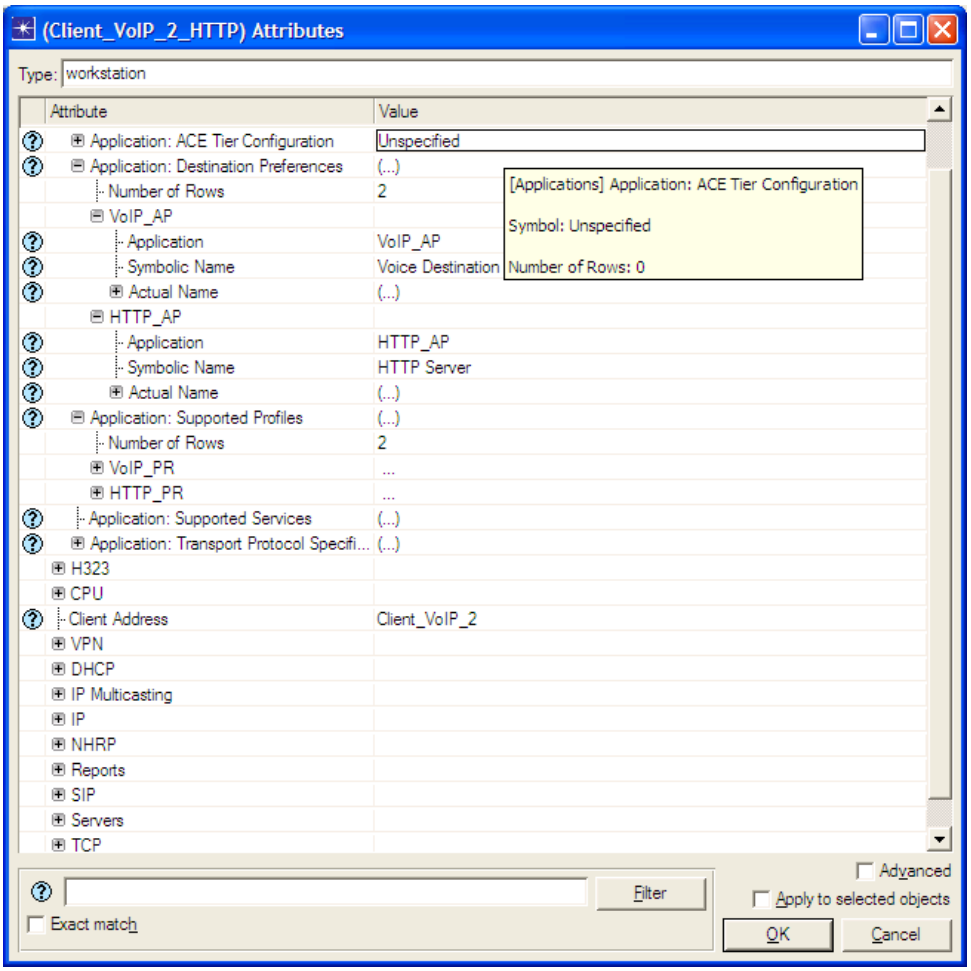


Obr. 2.3: Tabulka s definicí profilů v Profile_Config

Pro nastavení serveru je třeba vybrat aplikace a profily, které má daný server podporovat – v tomto případě se tedy jedná o aplikaci pojmenovanou jako FTP_AP a profil s názvem FTP_PR. Toho docílíme nastavením těchto parametrů v atributu *Application: Supported Profiles*, resp. *Application: Supported Services*.

Nastavení na klientské stanici je poněkud obtížnější. Zde se musí rozlišovat profily, které má stanice podporovat, služby, které má stanice nabízet, a určit cílová koncová zařízení, se kterými bude komunikace na daném profilu a službě probíhat. U zmíněného klienta je tedy třeba nastavit atribut *Application: Supported Profile* na VoIP a HTTP, atribut *Application: Supported Services* na VoIP, aby bylo možné tuto stanici zavolat a atribut *Application: Destination Preferences*, kde nastavíme partnerské stanice pro komunikaci – tedy HTTP server a jednu z klientských stanic, která podporuje VoIP (v tomto případě byla zvolena stanice Client_VoIP_2_HTTP). Pro ilustraci nastavení této stanice zobrazuje obr. 2.4.

Nakonfigurováním ostatních koncových zařízení je vytvořen funkční model sítě bez zajištění kvality služeb.



Obr. 2.4: Konfigurace klientské stanice

2.3.2 Scénář 2 – model sítě s QoS

Aby bylo možné později při simulacích provádět porovnání jednotlivých modelů sítě (bez QoS a s QoS), je v Opnet Modeleru nutné vytvářet scénáře, které toto umožňují. Po vytvoření duplicitního scénáře 1 je možné přejít k implementaci QoS do modelu sítě.

Vzhledem k tomu, že chceme, aby typy dat rozpoznávaly aktivní prvky sítě (tedy routery), budou se veškeré parametry nastavovat právě na jednotlivých routerech.

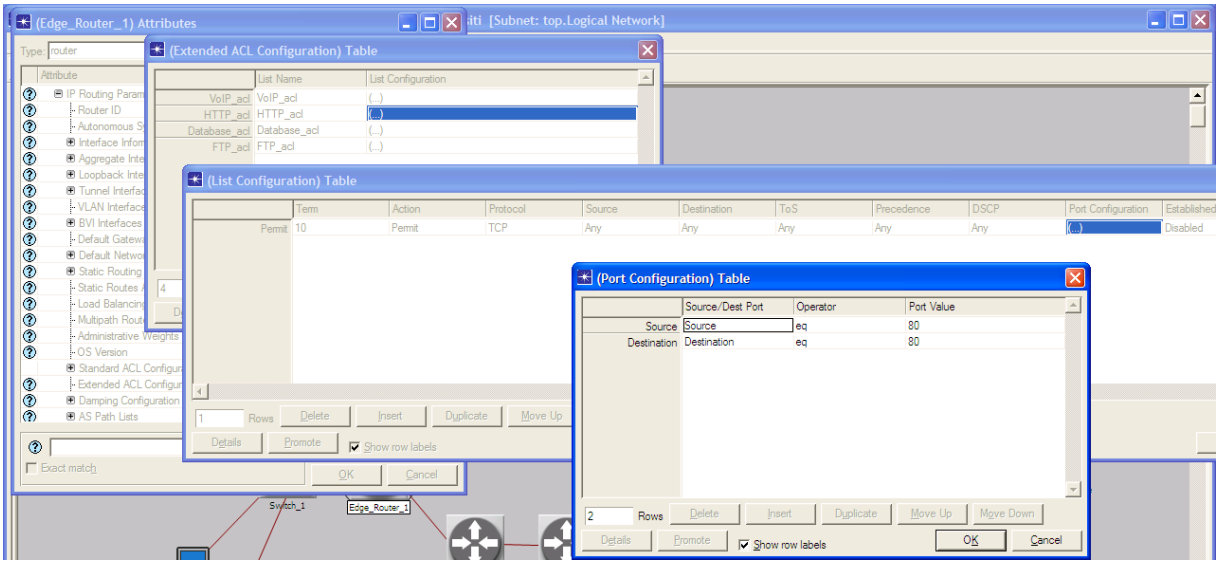
U hraničních směrovačů je třeba nastavit zacházení s příchozími a odchozími daty (ve vztahu k DiffServ doméně). Pro příchozí data do DiffServ domény nastavíme pravidla, podle kterých bude docházet k jejich značkování. To se provádí v atributu *IP -> IP QoS Parameters -> Traffic Policies*, kde se vytváří pravidla. V tomto případě stačí vytvořit pravidlo jedno, kde jednotlivým třídám služeb budou přiřazeny odpovídající značky DSCP. Pro používané třídy služeb byly nastaveny značky služeb dle tabulky 2.1.

Tab. 2.1: Hodnoty DSCP jednotlivých služeb

Typ třídy	**Hodnota ToS (DSCP)**
VoIP	EF
http	AF31
Databáze	AF21
FTP	AF11

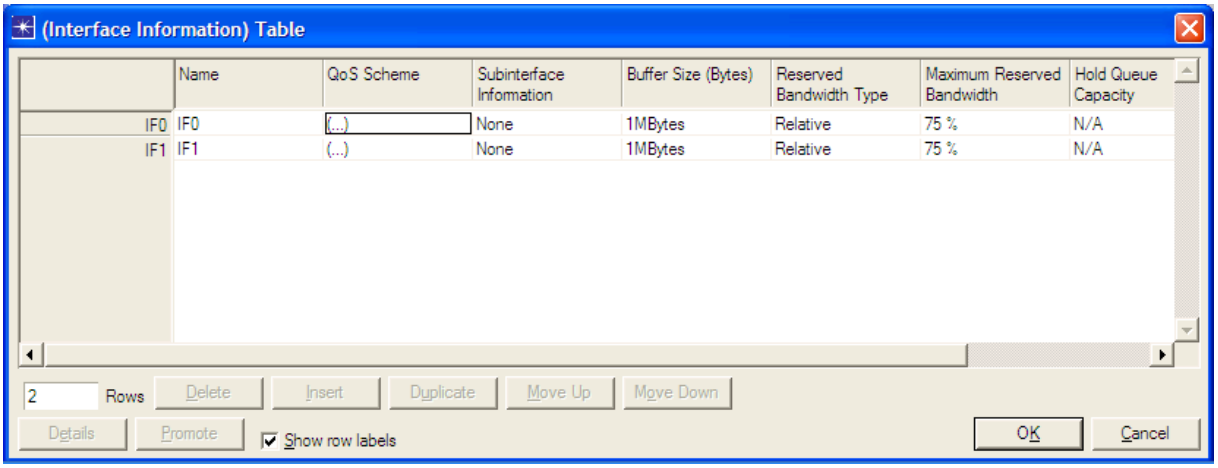
Třídy služeb jsou definovány v atributu *IP -> IP QoS Parameters -> Traffic Classes* a určují, jaké aplikace do dané třídy spadají. To se nastavuje v atributu *IP -> IP Routing Parameters -> Extended ACL Configuration*, kde se nastavují parametry pro třídění dat. Pro VoIP bylo nastaveno třídění dle protokolu UDP, pro HTTP dle protokolu TCP s číslem portu 80, pro FTP opět dle protokolu TCP, ovšem s číslem portu 20 a 21 a konečně pro databázi dle protokolu TCP s číslem portu 101. Toto číslo portu přiděluje ve výchozím nastavení Opnet Modeler a bylo zjištěno při simulacích pomocí debuggeru.

Jakmile jsou nadefinována pravidla, třídy a parametry ACL, je možné přistoupit ke konfiguraci jednotlivých portů směrovače. To je možné provést v atributu *IP -> IP QoS Parameters -> Interface Information*, kde jednotlivým portům přiřadíme schéma QoS (viz. obr. 2.5). Na neoznačené pakety budou aplikována vytvořená pravidla, označené pakety dle DSCP pak budou řazeny do fronty typu WFQ (Class Based).



Obr. 2.5: Konfigurace ACL na hraničních směrovačích

Vnitřní směrovače není nutné tak složitě konfigurovat, jelikož pracují s již označenými pakety dle hodnoty DSCP. Lze tedy použít výchozí nastavení pro řazení do fronty podle DSCP a řízené odesílání pomocí WFQ (class based). Nastavení se tedy provede obdobně jako u hraničního směrovače v atributu *IP -> IP QoS Parameters -> Interface Information* (viz. obr. 2.6).



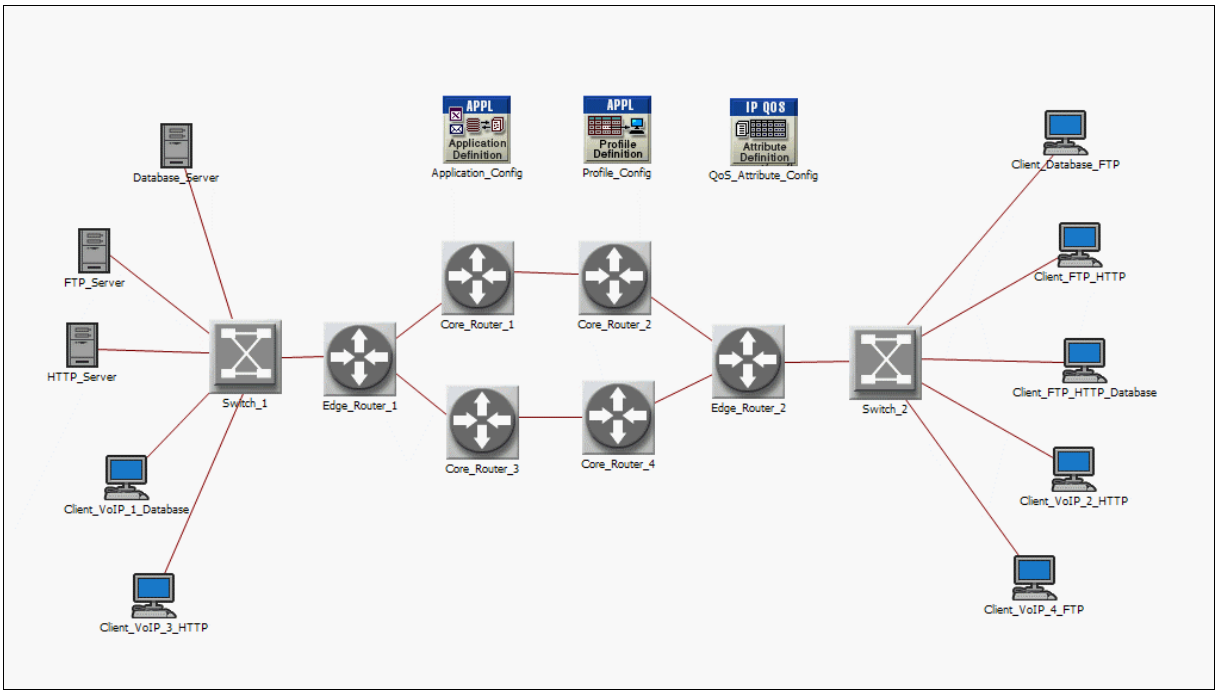
Obr. 2.6: Nastavení aktivních portů na vnitřním směrovači

Po nastavení zbývajících vnitřních routerů a hraničního směrovače, je základní model DiffServ domény připraven k simulaci.

2.3.3 Scénář 3 – model sítě s QoS a s alternativní trasou v DiffServ doméně

Scénář 3 vychází ze scénáře 2, proto byla vytvořena jeho kopie a upravena do podoby, kterou zobrazuje obr. 2.7. V hraničních směrovačích byly nakonfigurovány nové porty, jejichž zapojení bylo zjištěno pomocí kontextové nabídky položkou *Edit ports*. Nastavení schématu QoS u portu je shodné s nastavením ve scénáři 2.

Nové vnitřní směrovače mají totožnou konfiguraci jako staré.

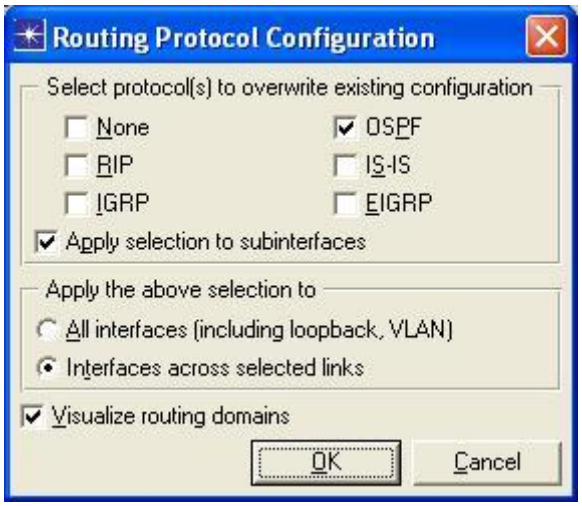


Obr. 2.7: Schéma modelu sítě s QoS a s alternativní trasou v DiffServ doméně (scénáře 3, 4, 5 a 6)

2.3.4 Scénář 4 – model DiffServ domény s OSPF

Podobně jako v předchozích případech byl vytvořen duplicitní scénář, který je následně upravován. Toho je docíleno tak, že v hlavním menu se klikne na *Scenarios -> Duplicate Scenario.*

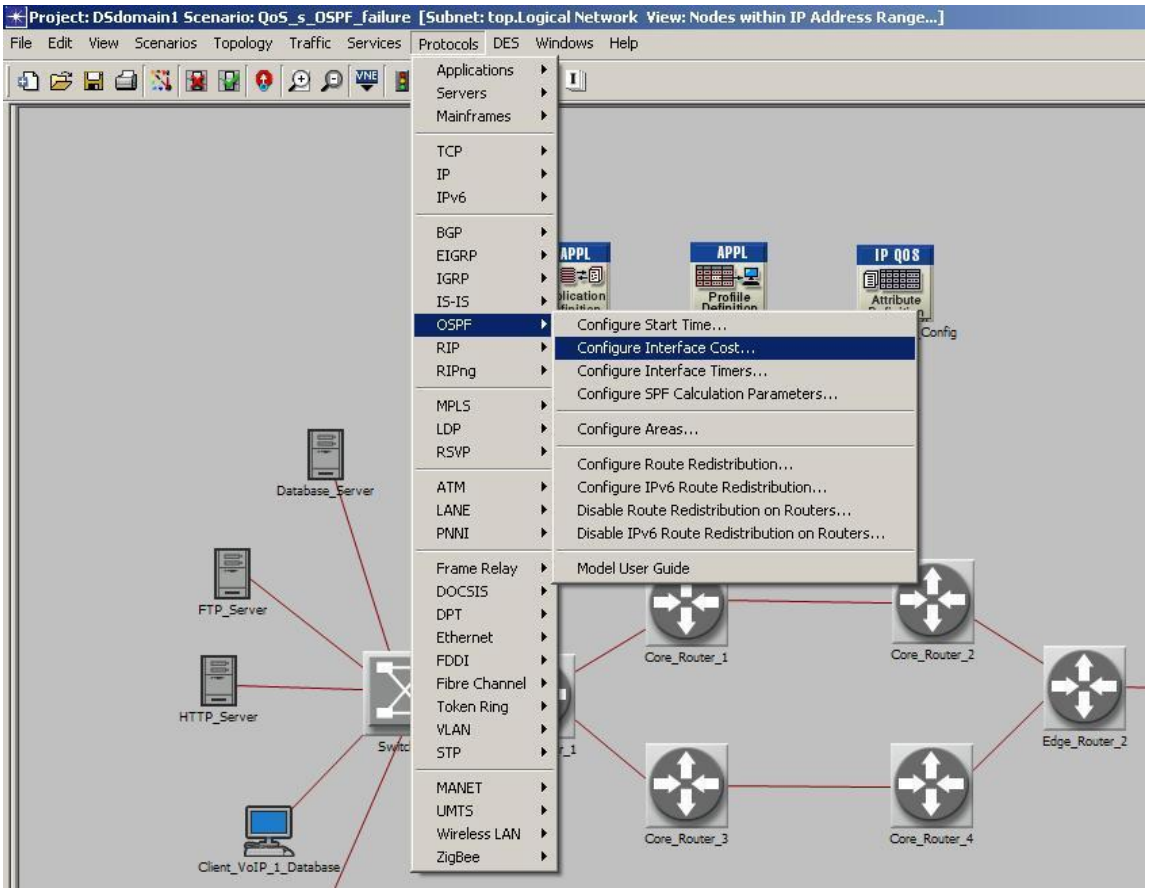
Do nově vzniklého scénáře byl v DiffServ doméně implementován směrovací protokol OSPF. Nejdříve byla vybrána rozhraní, na kterých bude směrovací protokol nakonfigurován – to se provádí podržením klávesy *CTRL* a vybráním linek, mezi kterými má být směrovací protokol OSPF funkční. Poté je třeba kliknout na *Protocols -> IP -> Routing -> Configure Routing Protocols.* Zobrazí se okno *Routing Protocol Configuration,* kde je nutné nastavit protokol OSPF tak, jak je uvedeno na obr. 2.8.



Obr. 2.8: Nastavení protokolu OSPF

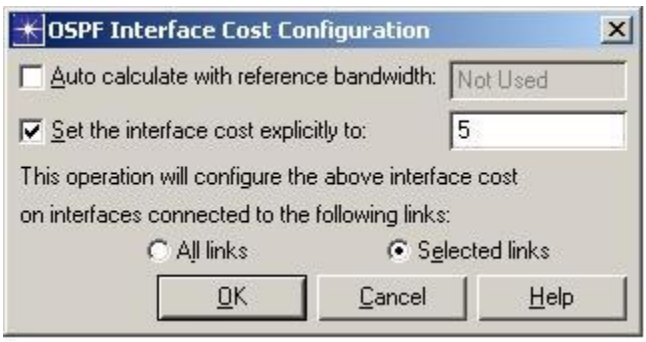
Poté je třeba jednotlivým cestám v DiffServ doméně nastavit cenu (metriku) tak, aby veškerá posílaná data přes DiffServ doménu putovala pouze jednou cestou. Obě trasy obsahují stejné prvky sítě, mají stejnou šířku pásma atd., takže se budou lišit pouze v nastavení rozdílné ceny, což zaručí možnost sledovat pouze skutečný vliv výpadku linky a žádné další nežádoucí parametry.

Konfigurace ceny se provádí kliknutím na položku *Protocols* v hlavním menu, vybráním protokolu *OSPF* a zvolením položky *Configure Interface Cost*, jak znázorňuje obr. 2.9. Vzhledem k tomu, že zmíněné nastavení se týká konkrétních linek – spojů mezi jednotlivými směrovači - je nejdříve nutné vybrat linky, kterých se konfigurace týká.



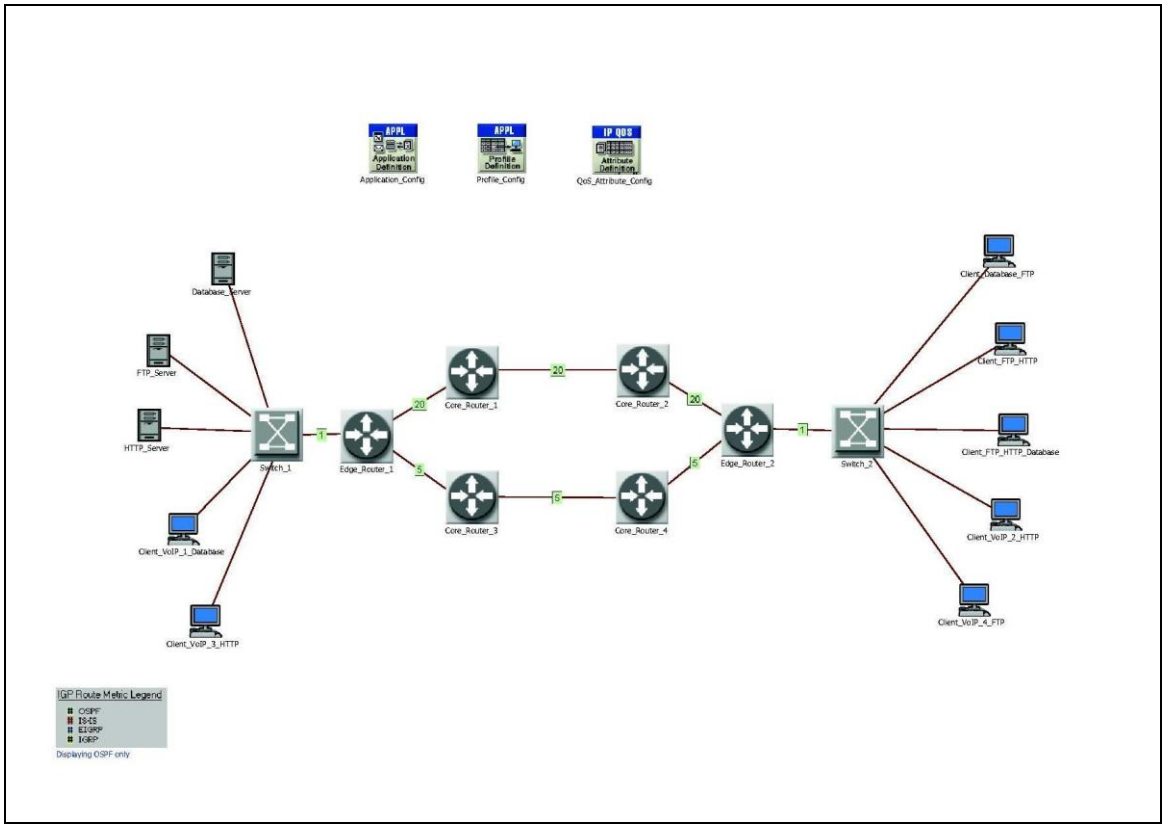
Obr. 2.9: Nalezení konfigurace ceny linky u OSPF

Obr. 2.10 znázorňuje nastavení ceny linky. Je nutné vybrat možnost *Set the interface cost explicitly to:*, aby bylo možné nastavit vlastní cenu. V rámci tohoto projektu byly nastaveny ceny na hodnotu 5 pro linky na spodní cestě DiffServ domény a na hodnotu 20 pro linky na horní cestě DiffServ domény. Jelikož je konfigurována cena jen pro určité linky, je třeba v konfiguraci také nastavit *Selected links.*



Obr. 2.10: Konfigurace ceny linky

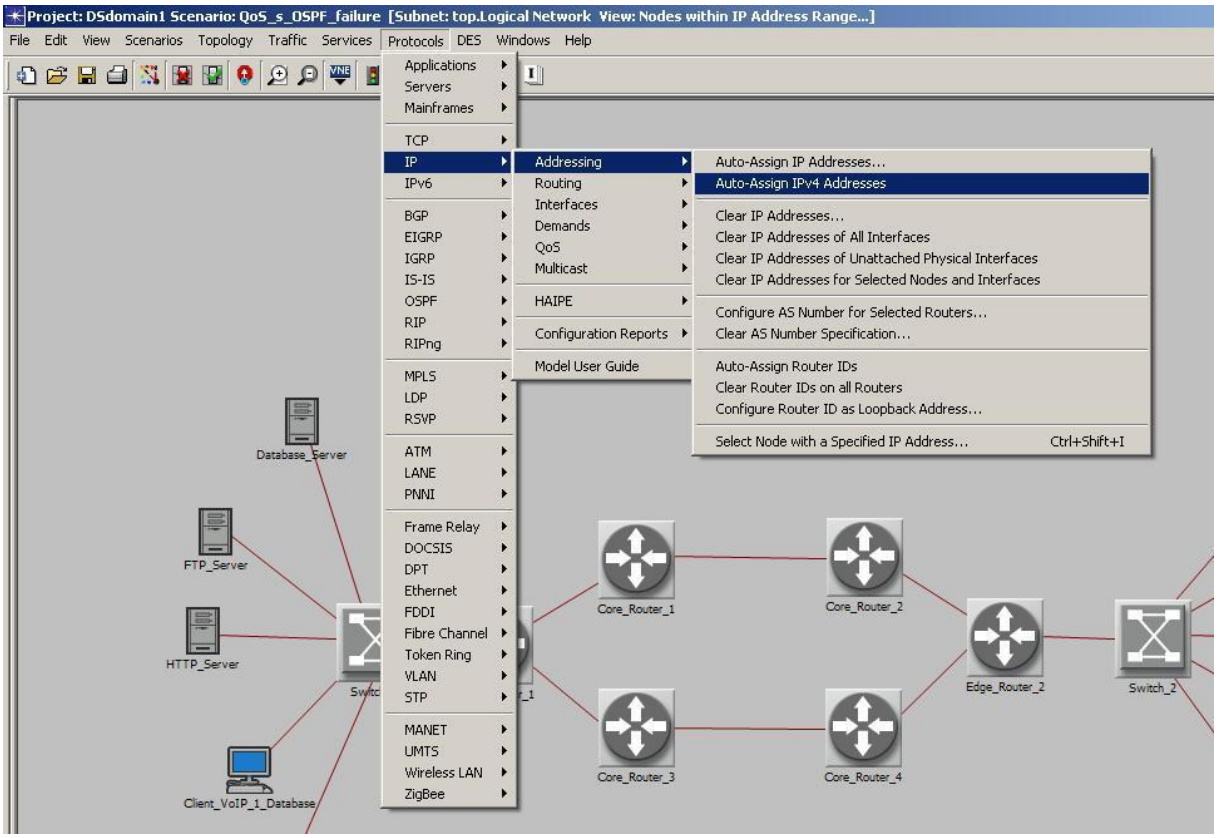
Z výše uvedeného je tedy patrné, že v DiffServ doméně tvoří hlavní trasu – s nižší cenou (metrikou) – spodní trasa, tedy přes vnitřní směrovač *Core_Router_3* a *Core_Router_4* a alternativní trasu horní trasa, tedy přes vnitřní směrovač *Core_Router_1* a *Core_Router_2*. Správnost nastavení byla ověřena kliknutím na *View -> Visualise IP Routing -> Costs* – správná konfigurace je tedy patrna z obr. 2.11.



Obr. 2.11: Zobrazení cen jednotlivých tras

Po nakonfigurování cen je vhodné ověřit nastavení časovačů OSPF protokolu, které se provádí v položce *Protocols -> OSPF -> Configure Interface Timers*. Opnet používá standardní, výchozí hodnoty časovačů pro LAN sítě tak, jak jsou uváděny v RFC2328 [16], tedy interval rozesílání Hello paketů každých 10s a od této hodnoty odvozený Dead interval 40s (čtyřnásobek Hello paketu).

Dále bylo nutné přidělit jednotlivým směrovačům IP adresy, aby směrování a plnění směrovacích tabulek pracovalo korektně. To bylo provedeno opět v položce hlavního menu - *Protocols* - jak je zobrazeno na obr. 2.12.



Obr. 2.12: Automatické přiřazení IP adres

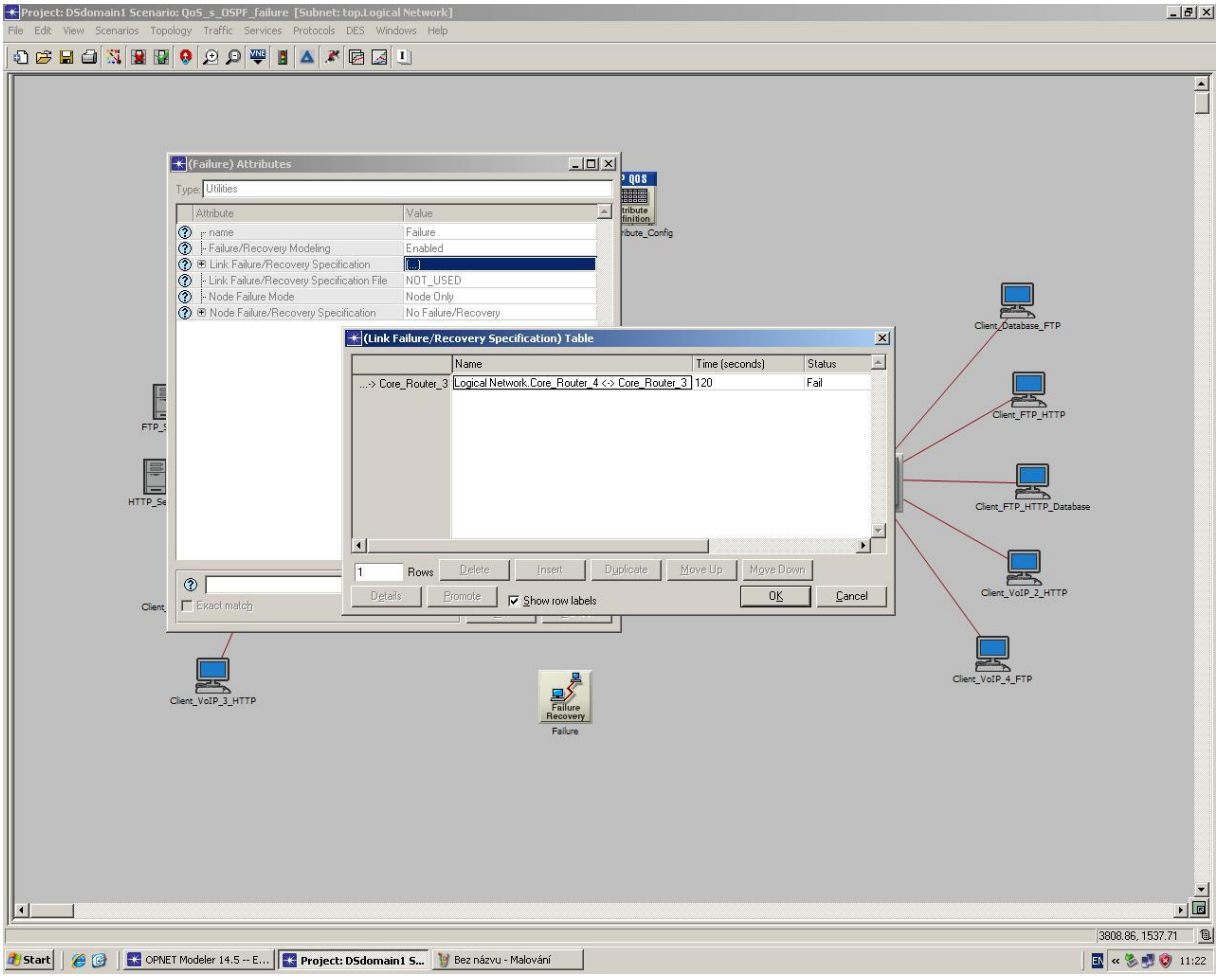
Správnost nastavení protokolu OSPF a cen jednotlivých linek byla ověřena vyexportováním směrovacích tabulek a jejich kontrolou. To bylo provedeno kliknutím na *Protocols -> IP -> Routing -> Export Routing Table.* Směrovací tabulky hraničních směrovačů a vnitřních směrovačů jsou uvedeny v příloze.

2.3.5 Scénář 5 – DiffServ doména s výpadkem linky

Tento scénář vychází opět z předchozího scénáře a byl do něj doplněn pouze objekt simulující výpadek linky. V Opnetu je výpadek linky simulován pomocí objektu *Failure_Recovery*, který se nachází v paletě objektů.

Stavy linek se tedy editují v atributech tohoto objektu a spočívají ve výběru konkrétní linky, konfiguraci stavu linky, do které má linka přejít (výpadek/obnovení), a konfiguraci času, kdy má ke změně stavu dané linky dojít.

Obr. 2.13 tedy znázorňuje konfiguraci výpadku linky na hlavní cestě v DiffServ doméně. K výpadku linky dojde na spojení mezi vnitřními směrovači *Core_Router_3* a *Core_Router_4* po 600 sekundách od začátku spuštění simulace.



Obr. 2.13: Ilustrace konfigurace výpadku linky

2.3.6 Scénář 6 – DiffServ doména s výpadkem linky a jejím následném obnovení

V průběhu vytváření tohoto projektu byla položena otázka, jak asi model sítě zareaguje při obnovení provozu linky – přesněji řečeno, za jak dlouho dojde k přesměrování provozu zpět na dolní cestu s vyšší cenou a zda-li toto obnovení bude mít vliv na kvalitu služeb.

Za tímto účelem byl tedy vytvořen tento scénář vycházející s předcházejícího, ve kterém byla pouze v objektu Failure změněna konfigurace a pro pořádek byl tomuto objektu změněn název na Failure_Recovery. V atributech objektu byl přidán v tabulce *Link Failure/Recovery Specification* další řádek, ve kterém byla vybrána opět stejná linka, tedy *LogicalNetwork_Core_Router_4<->Core_Router_3*, nastaven čas na 900s a stav linky na *Recover*. Tím bude zajištěno, že po 900s od začátku simulace (a tedy po 300s výpadku

linky) dojde k opětovnému nahození linky a provoz by měl být přesměrován zpět na trasu s vnitřními směrovači *Core_Router_3* a *Core_Router_4*.

2.4 Konfigurace sledovaných charakteristik a simulace

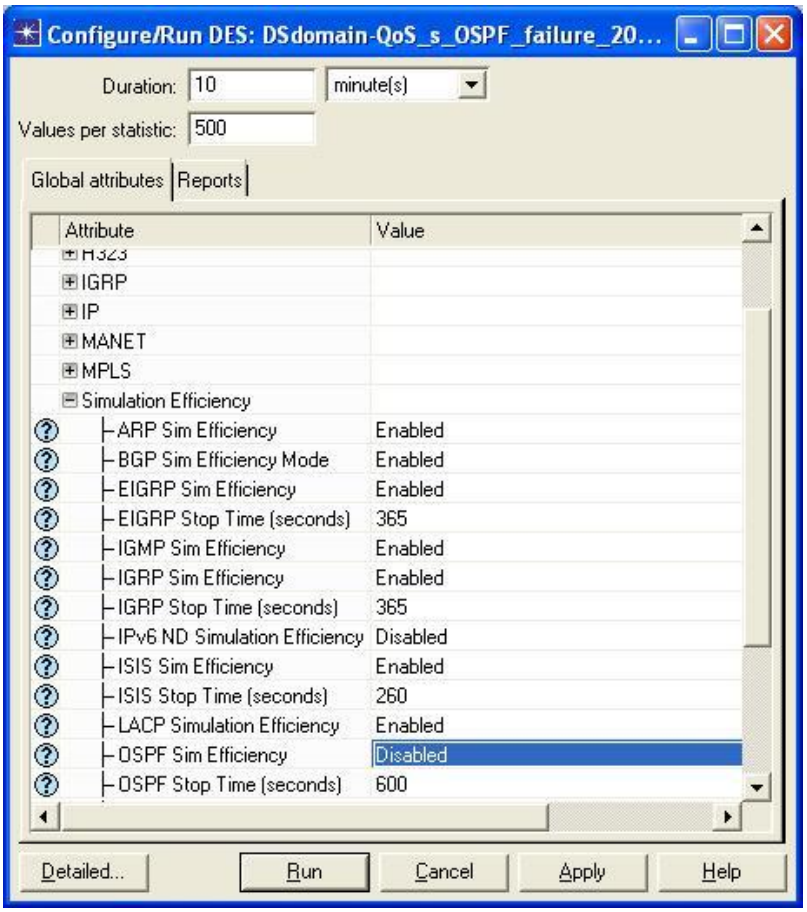
V Opnet Modeleru se pro každý scénář dají nastavit odlišné charakteristiky, které chceme sledovat, proto je nutné provést nastavení v každém scénáři. To se provádí kliknutím na položku *Choose Individual DES Statistics*, kde vybereme charakteristiky, které budeme sledovat.

V případě tohoto projektu byly vybrány charakteristiky jitteru a zpoždění, důležité pro VoIP, a dále charakteristiky provozu FTP, HTTP a databáze – jedná se o charakteristiky globální.

Dále budeme chtít sledovat charakteristiky na vnitřních směrovačích DiffServ domény zejména u scénáře 3, abychom zjistili, zda a jaký má vliv na kvalitu přenosu delší trasa (s větší metrikou). Konfiguraci provedeme označením objektu vnitřního směrovače a z kontextové nabídky opět vybereme položku *Choose Individual DES Statistic*, kde je třeba označit požadované statistiky.

Ve scénáři 4, 5 a 6 bude také podstatné sledovat provoz na jednotlivých cestách v DiffServ doméně, aby byla ověřena funkčnost výpadku linky, množství zahozených paketů při přenosu sítí a případně velikost využití zásobníků (bufferů) ve směrovačích pro další analýzy.

Opnet Modeler v rámci efektivity simulací ve výchozím nastavení vypíná po 180 sekundách od startu simulace rozesílání hello paketů a šíření informací o síti směrovacími protokoly, což je v případě tohoto projektu nežádoucí jev. Proto je nutné tuto funkci deaktivovat v nastavení simulací – konkrétně *DES -> Configure/Run Discrete Simulation -> Simplified*. Zde je nutné v záložce *Simulation Efficiency* nastavit hodnotu na *Disabled* (viz. obr. 2.14).



Obr. 2.14: Okno konfigurace simulace – vypnutí funkce Simulation Efficiency

Nastavení simulací celého projektu (tedy všech scénářů) se provádí v menu *Scenarios -> Manage Scenarios…*, kde ve sloupci *Results* zvolíme *<collect>* (případně *<recollect>*) pro každý scénář a nastavíme dobu simulace *Duration* na 30 minut. Samotná simulace se pak spustí kliknutím na *OK*.

Po dokončení simulace lze výsledky vyvolat kliknutím na položku *View Results* v kontextové nabídce pracovní plochy projektu.

2.5 Výsledky simulací

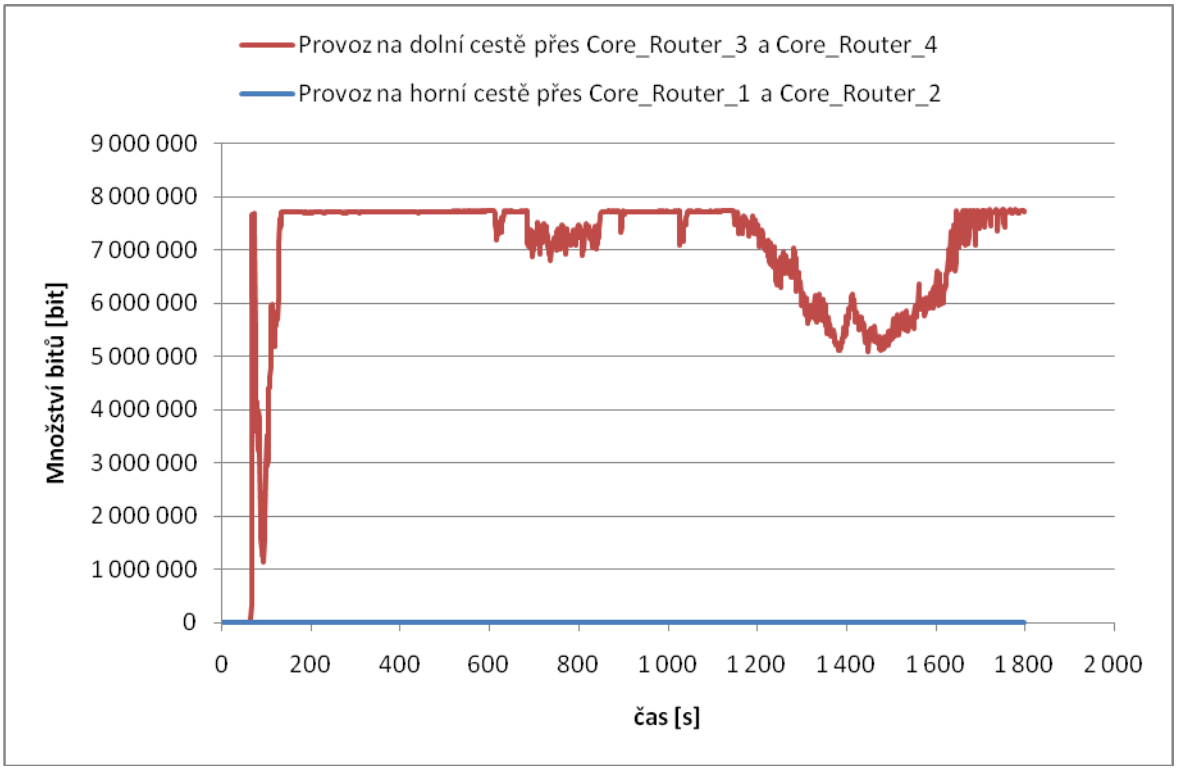
V průběhu vytváření modelu sítě s DiffServ doménou simulující výpadek linky a nalézání alternativní trasy byly prováděny průběžné simulace ověřující správnost konfigurace služeb a aktivní prvků sítě. Mezi tyto průběžné simulace patří průběhy ze scénářů 1 (základní model sítě), 2 (model sítě s QoS) a 3 (model sítě s QoS a alternativní trasou v DiffServ

doméně). Některé výsledky těchto simulací jsou uvedeny v příloze této práce, ostatní jsou dohledatelné přímo v projektu Opnet Modeleru na přiloženém CD.

V této kapitole budou tedy uvedeny a zanalyzovány průběhy týkající se zejména výpadku linky, jejich vlivu na kvalitu služeb a ověření správnosti akcí prováděných po výpadku linky. Tyto průběhy se nachází ve scénáři 4 (model sítě s QoS a OSPF), 5 (model sítě s výpadkem linky) a 6 (model sítě s výpadkem linky a jejím následném nahození).

2.5.1 Analýza výpadku linky a jejího opětovného nahození

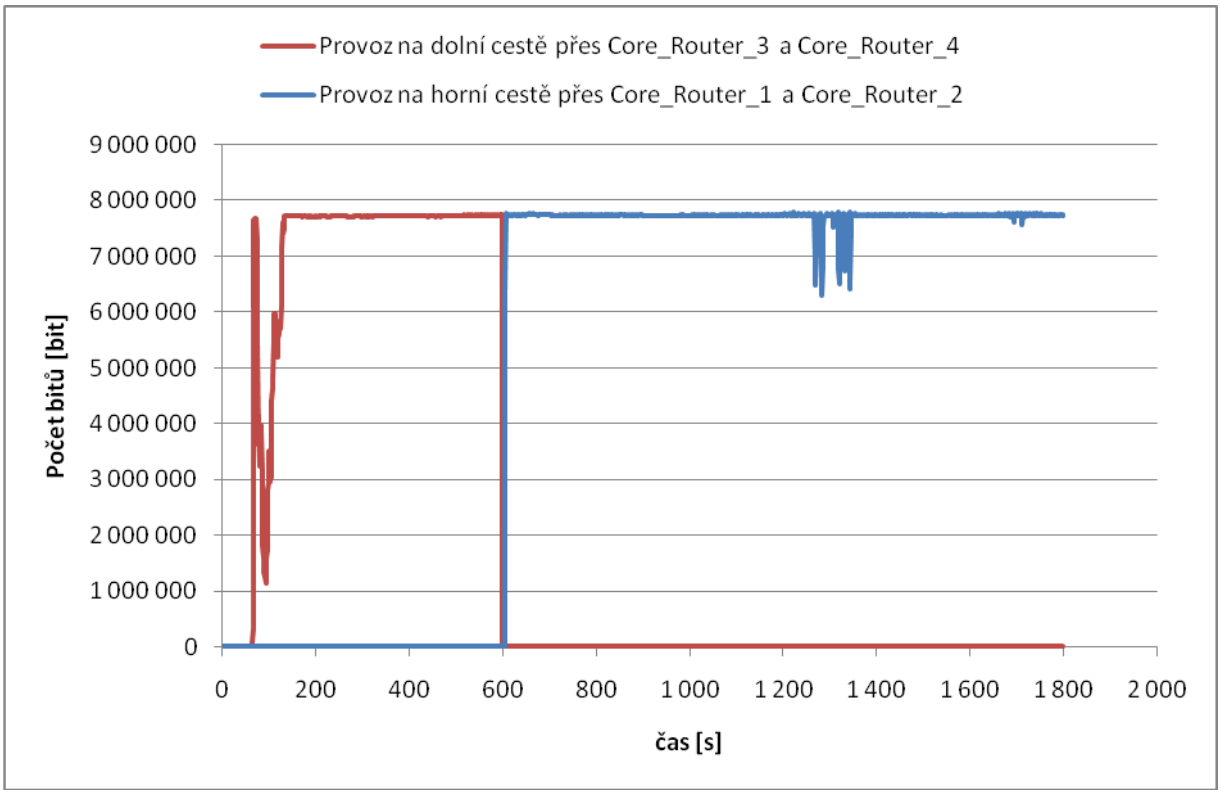
Po provedení simulací byla nejdříve ověřena správnost toku dat DiffServ doménou – tedy, že veškerý provoz v síti proudí DiffServ doménou přes vnitřní směrovač *Core_Router_3* a *Core_Router_4*, kde linky mají nastavenu vyšší cenu.



Obr. 2.15: Závislost počtu přenesených bitů na čase v jednotlivých cestách DiffServ domény

Z grafu uvedeném na obr. 2.15 je patrné, že konfigurace směrovacího protokolu OSPF byla provedena správně, jelikož veškerý provoz v DiffServ doméně skutečně probíhá cestou s vyšší cenou, tedy tou dolní, zatímco horní cestou netečou žádná data a není zde žádný provoz po celou dobu simulace.

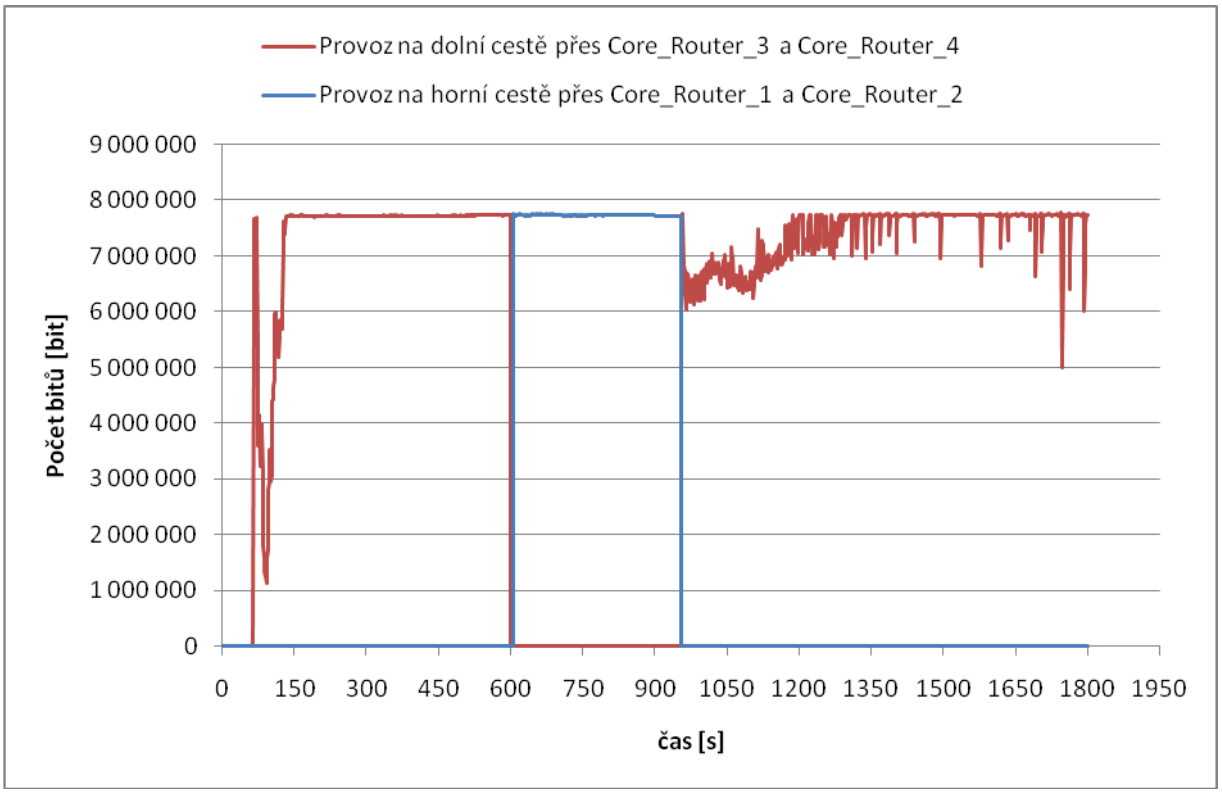
Následně byla ověřena správná funkčnost simulace výpadku linky. Opět byl sledován provoz na linkách mezi vnitřními směrovači, jejichž průběh je zobrazen na obr. 2.16.



Obr. 2.16: Závislost přenesených bitů na čase hlavní a alternativní trasou

Z grafu je jasně vidět, že v době 600s po začátku simulace dochází na hlavní trase k výpadku linky a k přesměrování provozu na alternativní, horní trasu s nižší cenou. Směrovací protokol OSPF tuto změnu stavu linky – výpadek – zaznamenává velmi rychle, takže lze očekávat, že vliv na kvalitu služeb v DiffServ doméně bude nepatrný.

Následující graf (viz. obr. 2.17) znázorňuje opět provoz na linkách mezi vnitřními směrovači DiffServ domény, tentokrát však ze scénáře 6, kdy je simulováno opětovné nahození linky a to v čase 900s po začátku simulace.



Obr. 2.17: Průběh závislosti přenesených bitů na čase hlavní a alternativní trasou

Z grafu je vidět, že průběh simulace je téměř totožný s průběhem z minulého scénáře až do času přibližně 900s od začátku simulace. V 600s opět dojde k výpadku linky a síť reaguje téměř okamžitě. V čase 900s od začátku simulace dojde k nahození linky, avšak síť přesměruje provoz na hlavní trasu až po přibližně 50s, tedy v čase cca. 950s. To je způsobeno konfigurací směrovacího protokolu, přesněji řečeno dobou vypršení hello intervalů a dead intervalů. Pokud bychom tyto parametry nastavili na polovinu, tak síť zkonverguje přibližně o polovinu rychleji. K přepojení provozu v tomto případě dojde až tehdy, kdy celá síť zkonverguje a všechny směrovače obdrží aktualizace směrovacích tabulek. Jinými slovy data jsou 50s po obnovení provozu na hlavní trase stále přenášena alternativní, záložní linkou a až poté dojde k přesměrování na hlavní trasu. Pokud by však došlo k výpadku alternativní trasy ve stejný čas, kdy je hlavní trasa obnovena, směrovací protokol OSPF by okamžitě přesměroval provoz na obnovenou hlavní trasu (tedy již v čase krátce po 900s).

Zmíněný jev pozdějšího přesměrování provozu se také projevuje na začátku všech zmíněných simulací, ve kterých je použito směrovacího protokolu OSPF, kdy si aktivní prvky sítě (směrovače) vyměňují informace o stavu linek v síti a naplňují si své směrovací tabulky. Proto začnou data sítí proudit až po přibližně 60s od začátku simulace. Před touto

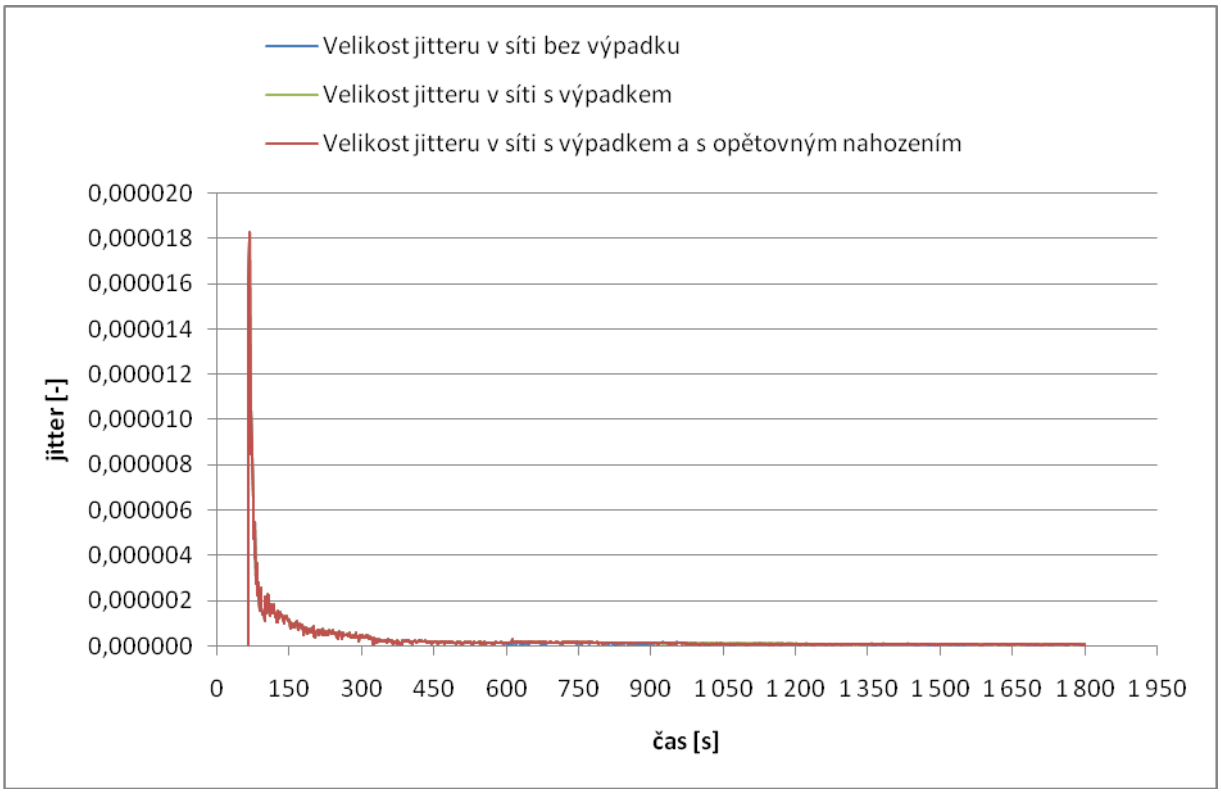
dobou nemají směrovače informace o umístění cílových a zdrojových adres, a proto nemohou data přenášet.

Na základě výše zmíněných poznatků lze konstatovat, že směrovací protokol OSPF je skutečně stavovým směrovacím protokolem. V případě výpadku hlavní trasy okamžitě reaguje přepojením na trasu s nižší cenou, avšak v případě, že data tečou sítí po určité trase a v síti se vyskytne jiná, výhodnější trasa (s vyšší cenou), přesměruje provoz na tuto novou hlavní trasu až tehdy, kdy síť zkonverguje.

2.5.2 Analýza vlivu výpadku linky na kvalitu služeb

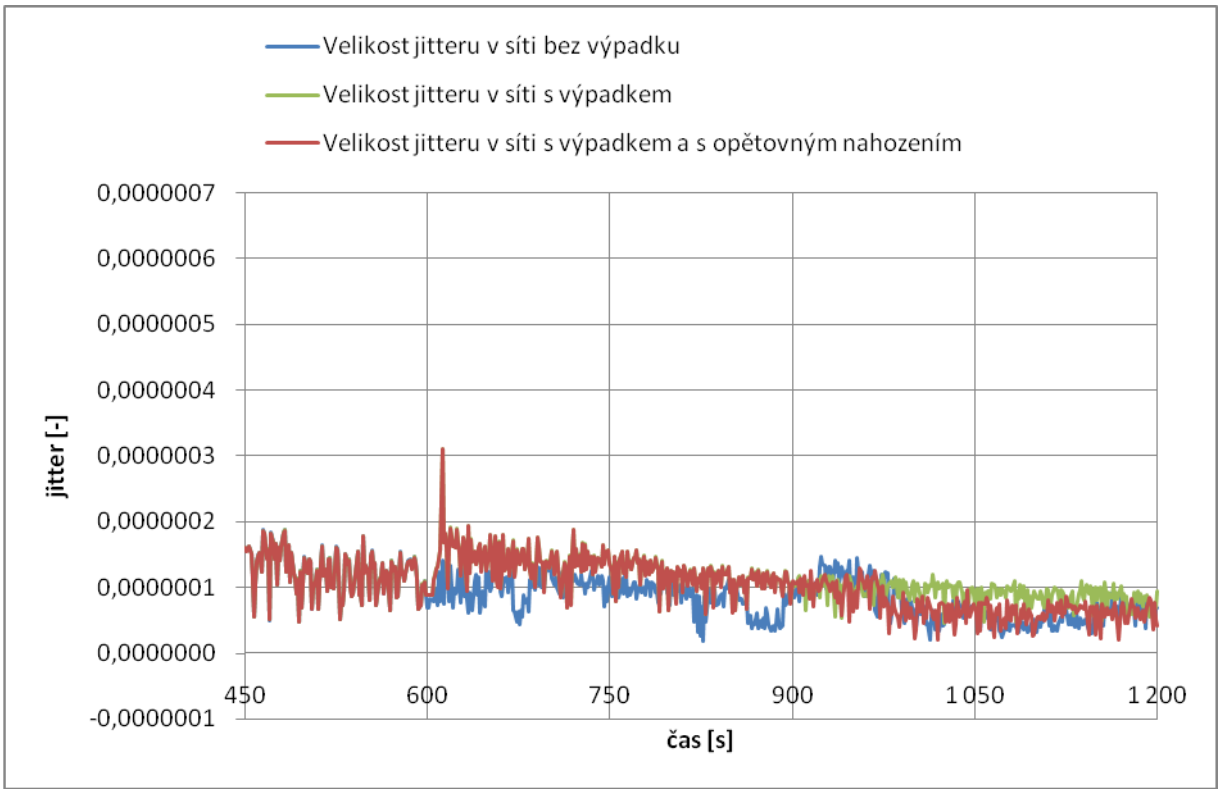
Po ověření správné funkčnosti simulací výpadku linky a jejího opětovného nahození byly provedeny analýzy týkající se vlivu výpadku linky na kvalitu služeb. Z již zmíněných poznatků lze očekávat, že výpadek linky by na kvalitu služeb měl mít minimální vliv.

První graf (viz. obr. 2.18) znázorňuje velikost jitteru v závislosti na čase. Z něj je patrné, že v průběhu času se velikost jitteru postupně snižuje až na přibližně konstantní hodnotu. Nejvyšší hodnota jitteru je tedy na počátku simulací, kdy dochází k sestavování spojení a zjišťování parametrů sítě.



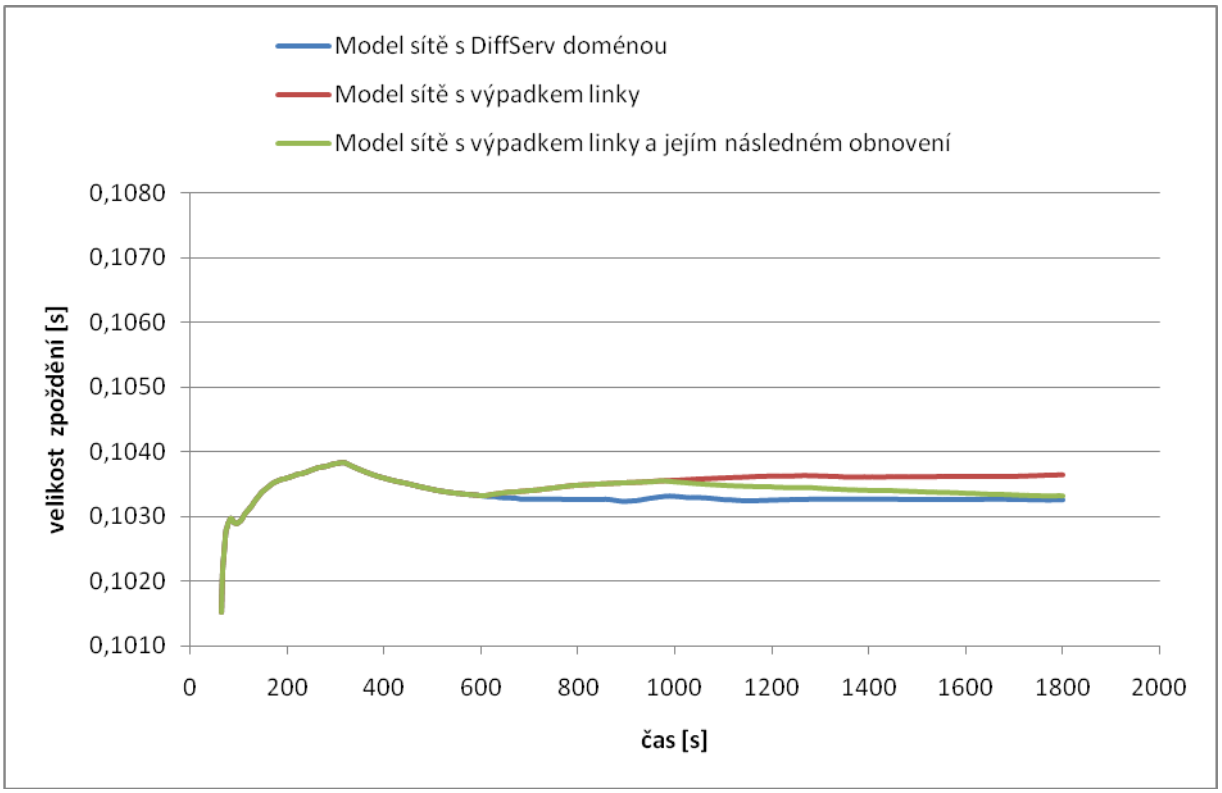
Obr. 2.18: Závislost velikosti jitteru na čase

Co se vlivu výpadku linky týče, tak jitter se změní dle očekávání jen nepatrně a po celý zbytek průběhu simulace udržuje konstantní, téměř nulový trend. Graf na obr. 2.19 znázorňuje zvětšený výřez předcházející charakteristiky, kde jsou změny jitteru zobrazeny výrazněji. Je vidět, že jitter nepatrně vzroste po výpadku linky, avšak v průběhu času dojde ke srovnání hodnot s hodnotami jitteru v síti bez výpadku. Obdobný trend, ovšem méně výrazný, má jitter v případě opětovného obnovení hlavní trasy. Z celkového pohledu však lze vyvodit závěr, že vliv výpadku linky na jitter je skutečně zanedbatelný, vezme-li v potaz, že maximální hodnota jitteru při výpadku linky dosahuje hodnoty lehce přes 0,3µs.



Obr. 2.19: Detail velikosti jitteru v závislosti na čase

Další graf (viz. obr. 2.20) znázorňuje průměrné zpoždění paketů v závislosti na čase mezi koncovými zařízeními sítě u hlasových přenosů. Výpadek linky a její opětovné nahození má také téměř zanedbatelný vliv na tento parametr a zpoždění se po celou dobu simulace pohybuje hluboko pod hodnotou 150ms, což je hodnota, kdy již přenášený hlas není srozumitelný. Z pohledu lidského vnímání by se tak sledované změny v síti u hlasového přenosu pomocí služby VoIP neměly vůbec projevit.



Obr. 2.20: Velikost zpoždění paketů mezi koncovými zařízeními u služby Voice

Průběhy dalších charakteristik jsou uvedeny v příloze. Jedná se o průběh množství zahozených paketů v závislosti na čase (viz. příloha 2), kdy je patrné, že k zahazování dochází na počátku simulace, než se naplní směrovací tabulky routerů a v době 600s, kdy dojde k výpadku linky na hlavní trase.

Dále jsou uvedeny velikosti jitteru celé sítě (příloha 3), na jedné z klientských stanic (příloha 4) a velikosti zpoždění přenášených dat (příloha 5). Jedná se o výsledky simulací prováděných v průběhu vytváření projektu a slouží k ověření správné konfigurace technologie DiffServ. Z grafů jasně vyplývá, že implementace QoS do sítí přináší efektivní využití síťových prostředků.

V přílohách 6 až 8 jsou pak pro úplnost zobrazeny průběhy množství přenesených dat v závislosti na čase u vybraných služeb a v příloze 1 jsou uvedeny výstupy směrovacích tabulek jednotlivých směrovačů, s jejichž pomocí byla provedena kontrola správné konfigurace cen protokolu OSPF.

ZÁVĚR

Cílem této diplomové práce bylo seznámit se s technologií rozlišovaných služeb, známou pod zkratkou DiffServ QoS, její strukturou a parametry, které ovlivňují kvalitu přenášených dat. Této problematice se věnuje první kapitola a na základě získaných znalostí pak byl v simulačním prostředí Opnet Modeler vytvořen model sítě, do kterého byla následně implementována technologie DiffServ.

Postupný vývoj modelu sítě naznačují jednotlivé scénáře v Opnet Modeleru, kdy byl nejdříve vytvořen jednoduchý model sítě simulující provoz základních služeb bez zajištění QoS.

Následně do modelu sítě byla implementována technologie DiffServ zabezpečující kvalitu služeb v DiffServ doméně. Byly tedy určeny hraniční a vnitřní směrovače a byla vytvořena pravidla stanovující značkování paketů přicházejících do DiffServ domény a pravidla zacházení s označenými pakety na vnitřních směrovačích.

V dalším kroku byla do modelu sítě již funkční DiffServ domény přidána alternativní trasa.

Po ověření správné funkčnosti vytvořeného modelu byl do DiffServ domény implementován směrovací protokol OSPF sledující stavy vybraných linek a v případě výpadku linky byla pomocí tohoto protokolu nalezena alternativní trasa.

Jakmile byl vytvořen zmíněný funkční model sítě s DiffServ doménou, byly provedeny analýzy na základě simulací výpadku linky na hlavní trase v DiffServ doméně.

Ze získaných výsledků vyplývá, že výpadek linky nemá zásadní vliv na přenášená data DiffServ doménou a kvalitu služeb. Hodnota jitteru se změní velmi nepatrně a velmi rychle se ustálí zpět na hodnotách jitteru ve scénáři, ve kterém nebyl simulován výpadek linky. Obdobně je tomu i u paketového zpoždění služby VoIP, kde však ustálení zpoždění neprobíhá tak rychle, avšak je třeba zdůraznit, že výkyv zpoždění se pohybuje v řádu jednotek milisekund, což je z pohledu praktického využití zanedbatelná hodnota, jelikož lidské ucho tento výkyv nezaznamená.

Vliv výpadku linky na QoS jednoznačně závisí na vlastnostech a typu použitého směrovacího protokolu. Protokol OSPF se na základě provedených simulací jeví jako vhodná volba a je jednoznačně vhodné jej používat i v rozsáhlejších sítích. Lze však očekávat, že

v rozsáhlejších sítích bude mít výpadek linky na kvalitu služeb větší vliv. Vše se odvíjí od velikosti sítě či oblasti autonomního systému (v případě této práce DiffServ domény) a správné konfigurace směrovacího protokolu tak, aby konvergence sítě byla co možná nejrychlejší.

SEZNAM LITERATURY A POUŽITÝCH ZDROJŮ

[1] GRYGÁREK, Petr. *Směrovací protokol OSPF* [online]. [2004] , změněno 1. listopadu 2004 [cit. 2009-04-23]. Kódováno v ISO-8859-2. Text v češtině. Dostupný z WWW: <http://www.cs.vsb.cz/grygarek/SPS/lect/OSPF/ospf.html>.

[2] HABRMAN, Robert. *Síťové protokoly (VI. část) - dynamické směrování mezí sítěmi: Dynamické směrování mezí sítěmi* [online]. 2007 , 11.10.2007 [cit. 2009-04-23]. Kódováno ve windows-1250. Vydáváno týdně. Text v češtině. Dostupný z WWW: <http://www.owebu.cz/pc-site/vypis.php?clanek=1267>.

[3] HABRMAN, Robert. *Síťové protokoly (XXI. část) – směrovací protokoly, Protokol OSPF* [online]. 2007 , 11.10.2007 [cit. 2009-04-23]. Kódováno ve windows-1250. Vydáváno týdně. Text v češtině. Dostupný z WWW: < http://www.owebu.cz/pc-site/vypis.php?clanek=1596>.

[4] JHA, Sanjay. Engineering Internet QoS. London: Artech House Publishers, 2002, ISBN: 1-58053-341-8

[5] JUNSEOK HWANG, RAJESH REVURU: Inter-Domain Diffserv Dynamic Provisioning and Interconnection Peering Study using Bandwidth Management Point -- A Simulation Evaluation, Syracuse University, ISBN: 1-56555-270-9

[6] KACÁLEK, Jan. *Modely pro zajištění kvality služeb IP sítích* [online]. c2006 [cit. 2008-12-16]. Dostupný z WWW: <http://amarok.cesketelekomunikace.cz/xkacal00/index.php?action=whatis>.

[7] MARCHESE, Mario. QoS OVER HETEROGENEOUS NETWORKS, 2007, John Wiley & Sons Ltd, ISBN 978-0-470-01752-4

[8] MOLNÁR, Karol. *VUT FEKT - Karol Molnár: MMOS - Moderní síťové technologie* [online]. [2007] [cit. 2008-12-16]. Dostupný z WWW: <http://www.utko.feec.vutbr.cz/~molnar/mmos/QoS.pdf>.

[9] MOLNÁR, Karol, ZEMAN, Otto, SKOŘEPA, Michal. *Moderní síťové technologie: Laboratorní cvičení* [online]. 2008 [cit. 2008-12-16]. Dostupný z WWW: <http://www.utko.feec.vutbr.cz/~molnar/mmos/MMOS_lab.pdf>.

[10] MOY, John. *RFC2328 : OSPF Version 2* [online]. 1998 [cit. 2009-04-23]. Dostupný z WWW: <http://docstore.mik.ua/rfc/rfc2328.html>.

[11] PETERKA, Jiří. Šířka pásma a její dělení. *Computerworld: e-archiv Jiřího Peterky* [online]. 1991, ročník 91, č. 43 [cit. 2008-12-16]. Dostupný z WWW: <http://www.earchiv.cz/a91/a143c110.php3>.

[12] SATRAPA, Pavel: IPv6 -- Velmi podrobný popis vlastností a schopností nového protokolu IPv6, 2008, ISBN: 80-86330-10-9

[13] SZIGETI, Tim, HATTING, Christina. *End-to-End QoS Network Design: Quality of Service in LANs, WANs, and VPNs*. Indianapolis: Cisco Press, 2004. 768 s. ISBN 978-1-58705-176-0.

[14] UBIK, Sven. *QoS a diffserv - Úvod do problematiky*. [s.l.], 2000. 12 s., technická zpráva. Dostupný z WWW: <http://www.cesnet.cz/doc/techzpravy/2000-6/diffserv.pdf>.

[15] VOZŇÁK, Miroslav: QoS v sítích s technologií VoIP, 2004, technická zpráva, VŠB TU Ostrava, zn.: voz-TR01-2004

[16] WANG, Zheng: Internet QoS: Architectures and Mechanisms for Quality of Service. San Francisco: Morgan Kaufmann, 2001, ISBN: 1-55860-608-4.

SEZNAM POUŽITÝCH ZKRATEK, VELIČIN A SYMBOLŮ

ACL	- z angl. Access Control List, tedy seznam pro řízení přístupu. V případě QoS se jedná o soubor pravidel definující značení neoznačených paketů přicházejících na hraniční směrovač.
BA	- z angl. Behaviour Aggregate, způsob třídění paketů v síťových prvcích.
Buffer	- vyrovnávací paměť, je určena pro dočasné uchování dat před jejich přesunem na jiné místo (zpracováním).
CB WFQ	- z angl. Class-Based Weighted Fair Queuing, systém front založený na třídách s váženou spravedlivou obsluhou.
CBS	- z angl. Committed Burst Size, velikost garantovaného shluku, parametr určující kvalitu síťového provozu.
CIR	- z angl. Commited Information Rate, garantovaná průměrná přenosová rychlost.
CRITIC/ECP	- z angl. Critical and Emergency Call Processing, priorita typu služby pro zprávy posílané autorizovanými zdroji ze státních složek civilní obrany (původní účel).
CU	- z angl. Currently Unused, označení momentálně nevyužitých bitů v poli DS.
Datagram	- logické seskupení informace zaslané jako jednotka síťové vrstvy přes zařízení na přenos dat bez předchozího založení spojení. Primárními informačními jednotkami v síti Internet jsou datagramy IP.
DiffServ	- z angl. Differentiated Services, technologie rozlišovaných služeb zajišťující QoS v síti.
DSCP	- z angl. Differentiated Services Code Point, 6tibitová značka pole DS definující způsob zacházení s pakety ve frontě portu směrovače.
EBS	- z angl. Excess Burst Size, velikost nadměrného shluku, parametr určující kvalitu síťového provozu.
FIFO	- z angl. First-In-First-Out, typ systému front založeného na obsluze paketu v takovém pořadí, v jakém byly přijaty.
FQ	- z angl. Fair Queuing, systém front se spravedlivou obsluhou.
FTP	- z angl. File Transfer Protocol, protokol patřící do rodiny TCP/IP protokol. Je určen pro přenos souborů.

HTTP - z angl. HyperText Transfer Protocol, původně určen pro výměnu hypertextových dokumentů ve formátu HTML.

IETF - z angl. Internet Engineering Task Force, organizace zabývající se standardy TCP/IP a soubory internetových protokolů.

IGRP - z angl. Interior Gateway Routing Protocol, směrovací protokol vyvíjený firmou Cisco.

Jitter - parametr určující velikost zpoždění paketu při průchodu sítí způsobený aktivními prvky sítě.

LSA - z angl. Link State Advertisment, typ zprávy rozesílaný směrovacím protokolem OSPF při kontrole dostupných směrovačů.

MF - z angl. MultiField Classification, vícepoložkové způsob paketů v síťových prvcích.

MPLS - z angl. MultiProtocol Label Switching, technologie zajištění kvality služeb založená na přepojování na základě značek, dříve označováno jako IntServ.

OSPF - z angl. Open Shortest Path First, typ směrovacího protokolu.

Paket - blok spolehlivě přenášených informací počítačovou sítí.

PBS - z angl. Peak Burst Size, velikost maximálního shluku, parametr určující kvalitu síťového provozu.

PHB - z angl. Per-Hop Behavior, způsob zacházení s pakety.

PIR - z angl. Peak Information Rate, maximální okamžitá přenosová rychlost, parametr určující kvalitu síťového provozu.

PQ - z angl. Priority Queuing, prioritní systém front.

QoS - z angl. Quality of Services, kvalita služeb.

RIP - z angl. Routing Information Protocol, typ jednoduchého směrovacího protokolu.

SLA - z angl. Service Level Agreement, dohoda o úrovni služeb.

SPF - z angl. Shortest Path First, algoritmus užívaný u protokolu OSPF.

srTCM - z angl. single rate Three Color Marker, princip značení datového toku u TB.

TB - z angl. Token Bucket, mechanismus měření provozu.

TCP/IP - z angl. Transmission Control Protocol/Internet Protocol. Rodina protokolů obsahující sadu protokolů pro komunikaci v počítačové síti.

ToS - z angl. Type of Service, způsob značení paketu pro zajištění QoS,

	používaný u IntServ.
trTCM	- angl. two rates Three Color Marker, princip značení datového toku u TB.
UDP	- z angl. User Datagram Protocol, jeden z tzv. internetových protokolů pro přenos dat sítí. Nenabízí možnost potvrzování přenášených dat.
VoIP	- z angl. Voice over Internet Protocol, služba nabízející přenos hlasu v IP sítích.
WFQ	- z angl. Weighted Fair Queuing, systém front s váženou spravedlivou obsluhou.
WRR	- z angl. Weighted Round Robin, systém front s váženou cyklickou obsluhou.

SEZNAM PŘÍLOH

Příloha 1 – Směrovací tabulky směrovačů v DiffServ doméně

Tab.: Vnitřní router Core_Router_1

Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	Direct	0	0	192.0.1.1	Logical Network.Core_Router_1	IF0	0.000
192.0.2.0/24	Direct	0	0	192.0.2.2	Logical Network.Core_Router_1	IF1	0.000
192.0.3.0/24	OSPF 1	110	21	192.0.1.2	Logical Network.Edge_Router_1	IF0	61.982
192.0.4.0/24	OSPF 1	110	40	192.0.2.1	Logical Network.Core_Router_2	IF1	61.982
192.0.5.0/24	OSPF 1	110	36	192.0.1.2	Logical Network.Edge_Router_1	IF0	61.982
192.0.6.0/24	OSPF 1	110	25	192.0.1.2	Logical Network.Edge_Router_1	IF0	61.982
192.0.7.0/24	OSPF 1	110	30	192.0.1.2	Logical Network.Edge_Router_1	IF0	61.982
192.0.8.0/24	OSPF 1	110	35	192.0.1.2	Logical Network.Edge_Router_1	IF0	61.982
Gateway of last resort is	not set						

Tab.: Vnitřní router Core_Router_2

Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	OSPF 1	110	40	192.0.2.2	Logical Network.Core_Router_1	IF0	N/A
192.0.2.0/24	Direct	0	0	192.0.2.1	Logical Network.Core_Router_2	IF0	N/A
192.0.3.0/24	OSPF 1	110	36	192.0.4.1	Logical Network.Edge_Router_2	IF1	N/A
192.0.4.0/24	Direct	0	0	192.0.4.2	Logical Network.Core_Router_2	IF1	N/A
192.0.5.0/24	OSPF 1	110	21	192.0.4.1	Logical Network.Edge_Router_2	IF1	N/A
192.0.6.0/24	OSPF 1	110	35	192.0.4.1	Logical Network.Edge_Router_2	IF1	N/A
192.0.7.0/24	OSPF 1	110	30	192.0.4.1	Logical Network.Edge_Router_2	IF1	N/A
192.0.8.0/24	OSPF 1	110	25	192.0.4.1	Logical Network.Edge_Router_2	IF1	N/A
Gateway of last resort is	not set						

Tab.: Vnitřní router Core_Router_3

Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	OSPF 1	110	25	192.0.6.2	Logical Network.Edge_Router_1	IF0	64.837
192.0.2.0/24	OSPF 1	110	45	192.0.6.2	Logical Network.Edge_Router_1	IF0	64.837
192.0.3.0/24	OSPF 1	110	6	192.0.6.2	Logical Network.Edge_Router_1	IF0	64.837
192.0.4.0/24	OSPF 1	110	30	192.0.7.1	Logical Network.Core_Router_4	IF1	64.837
192.0.5.0/24	OSPF 1	110	11	192.0.7.1	Logical Network.Core_Router_4	IF1	64.837
192.0.6.0/24	Direct	0	0	192.0.6.1	Logical Network.Core_Router_3	IF0	0.000
192.0.7.0/24	Direct	0	0	192.0.7.2	Logical Network.Core_Router_3	IF1	0.000
192.0.8.0/24	OSPF 1	110	10	192.0.7.1	Logical Network.Core_Router_4	IF1	64.837
Gateway of last resort is	not set						

Tab.: Vnitřní router Core_Router_4

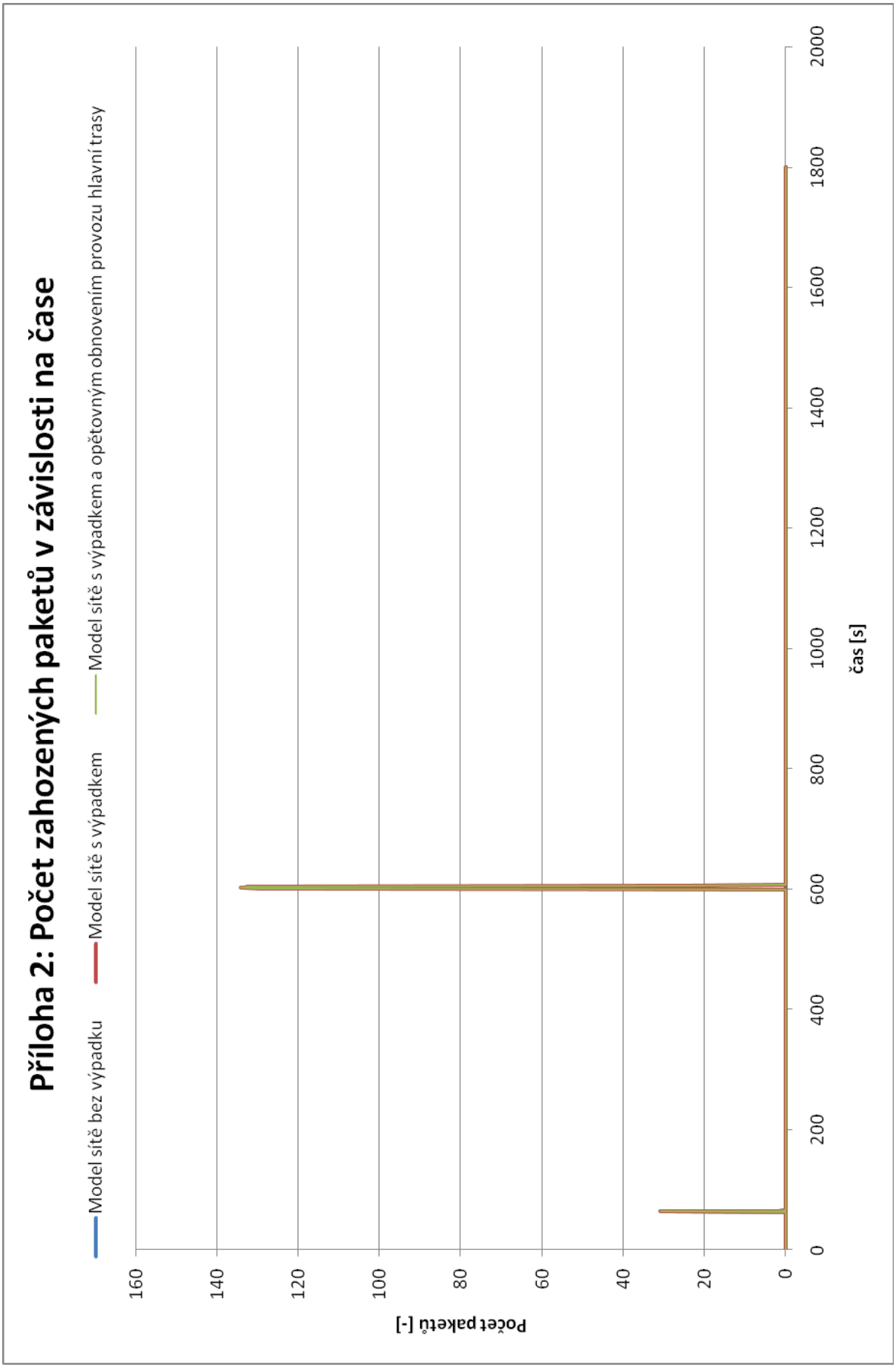
Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	OSPF 1	110	30	192.0.7.2	Logical Network.Core_Router_3	IF1	64.291
192.0.2.0/24	OSPF 1	110	45	192.0.8.2	Logical Network.Edge_Router_2	IF0	64.291
192.0.3.0/24	OSPF 1	110	11	192.0.7.2	Logical Network.Core_Router_3	IF1	64.291
192.0.4.0/24	OSPF 1	110	25	192.0.8.2	Logical Network.Edge_Router_2	IF0	64.291
192.0.5.0/24	OSPF 1	110	6	192.0.8.2	Logical Network.Edge_Router_2	IF0	64.291
192.0.6.0/24	OSPF 1	110	10	192.0.7.2	Logical Network.Core_Router_3	IF1	64.291
192.0.7.0/24	Direct	0	0	192.0.7.1	Logical Network.Core_Router_4	IF1	0.000
192.0.8.0/24	Direct	0	0	192.0.8.1	Logical Network.Core_Router_4	IF0	0.000
Gateway of last resort is	not set						

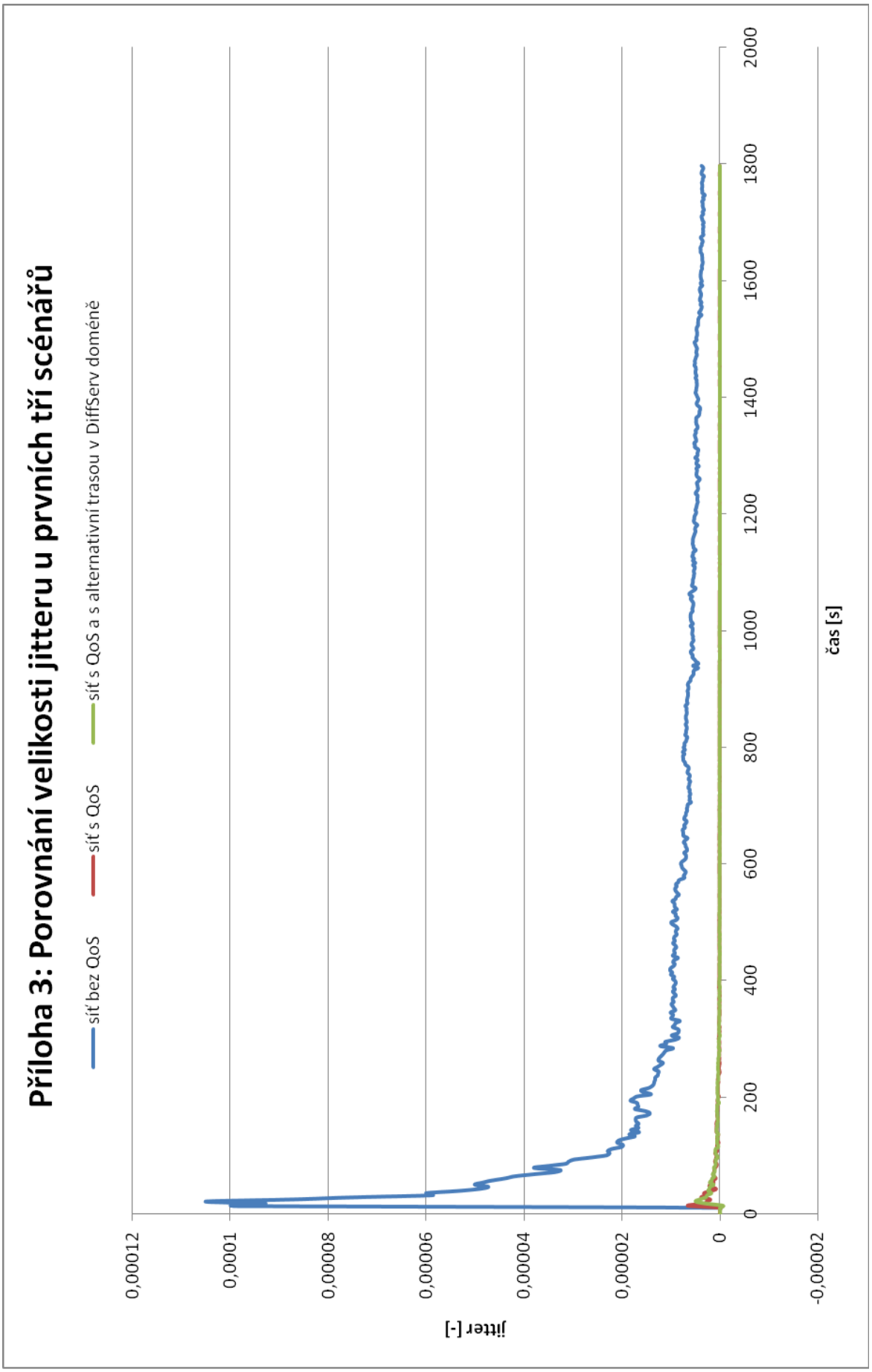
Tab.: Hraniční router Edge_Router_1

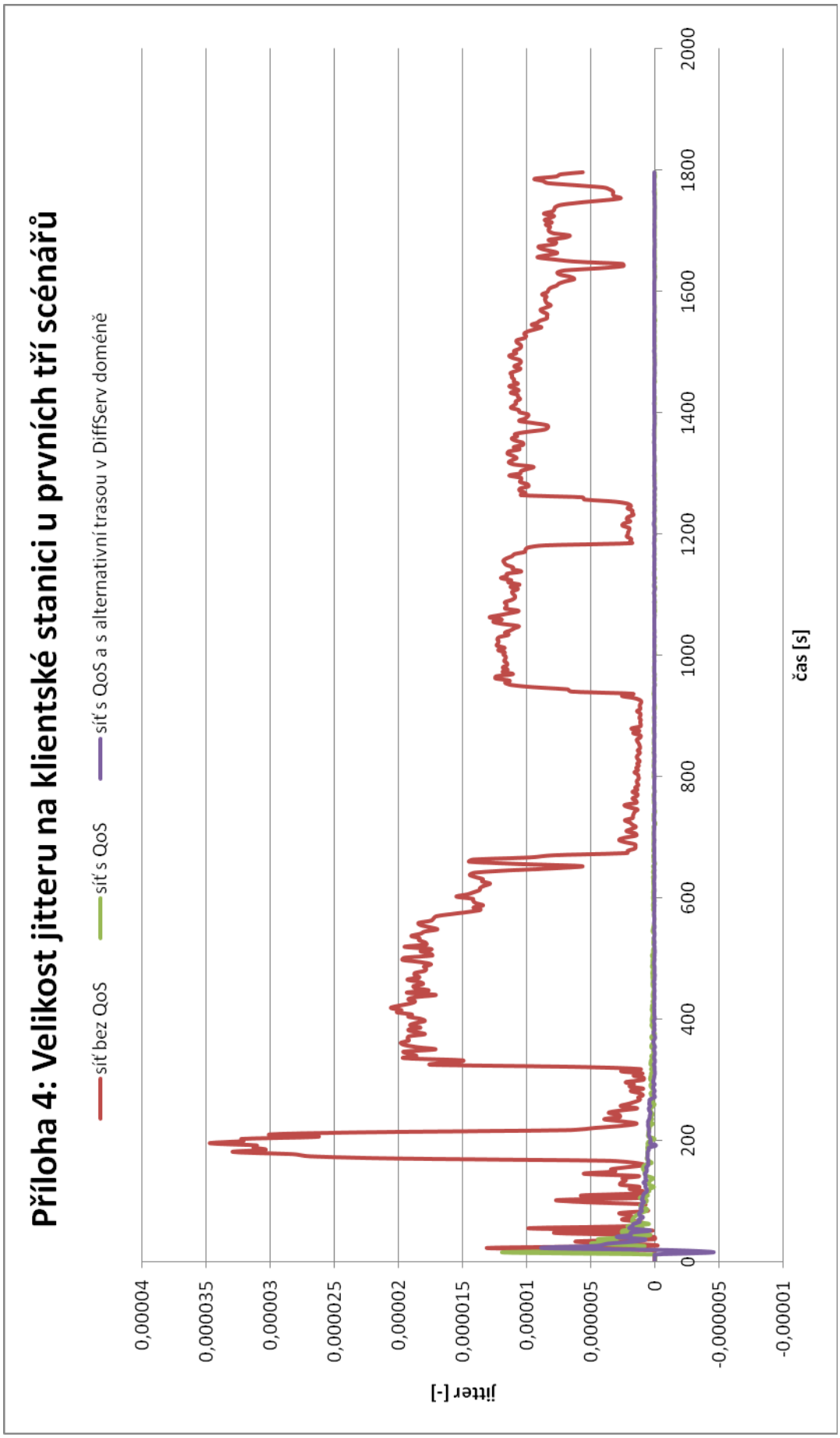
Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	Direct	0	0	192.0.1.2	Logical Network.Edge_Router_1	IF1	0.000
192.0.2.0/24	OSPF 1	110	40	192.0.1.1	Logical Network.Core_Router_1	IF1	65.352
192.0.3.0/24	Direct	0	0	192.0.3.2	Logical Network.Edge_Router_1	IF0	0.000
192.0.4.0/24	OSPF 1	110	35	192.0.6.1	Logical Network.Core_Router_3	IF2	65.352
192.0.5.0/24	OSPF 1	110	16	192.0.6.1	Logical Network.Core_Router_3	IF2	65.352
192.0.6.0/24	Direct	0	0	192.0.6.2	Logical Network.Edge_Router_1	IF2	0.000
192.0.7.0/24	OSPF 1	110	10	192.0.6.1	Logical Network.Core_Router_3	IF2	65.352
192.0.8.0/24	OSPF 1	110	15	192.0.6.1	Logical Network.Core_Router_3	IF2	65.352
Gateway of last resort is	not set						

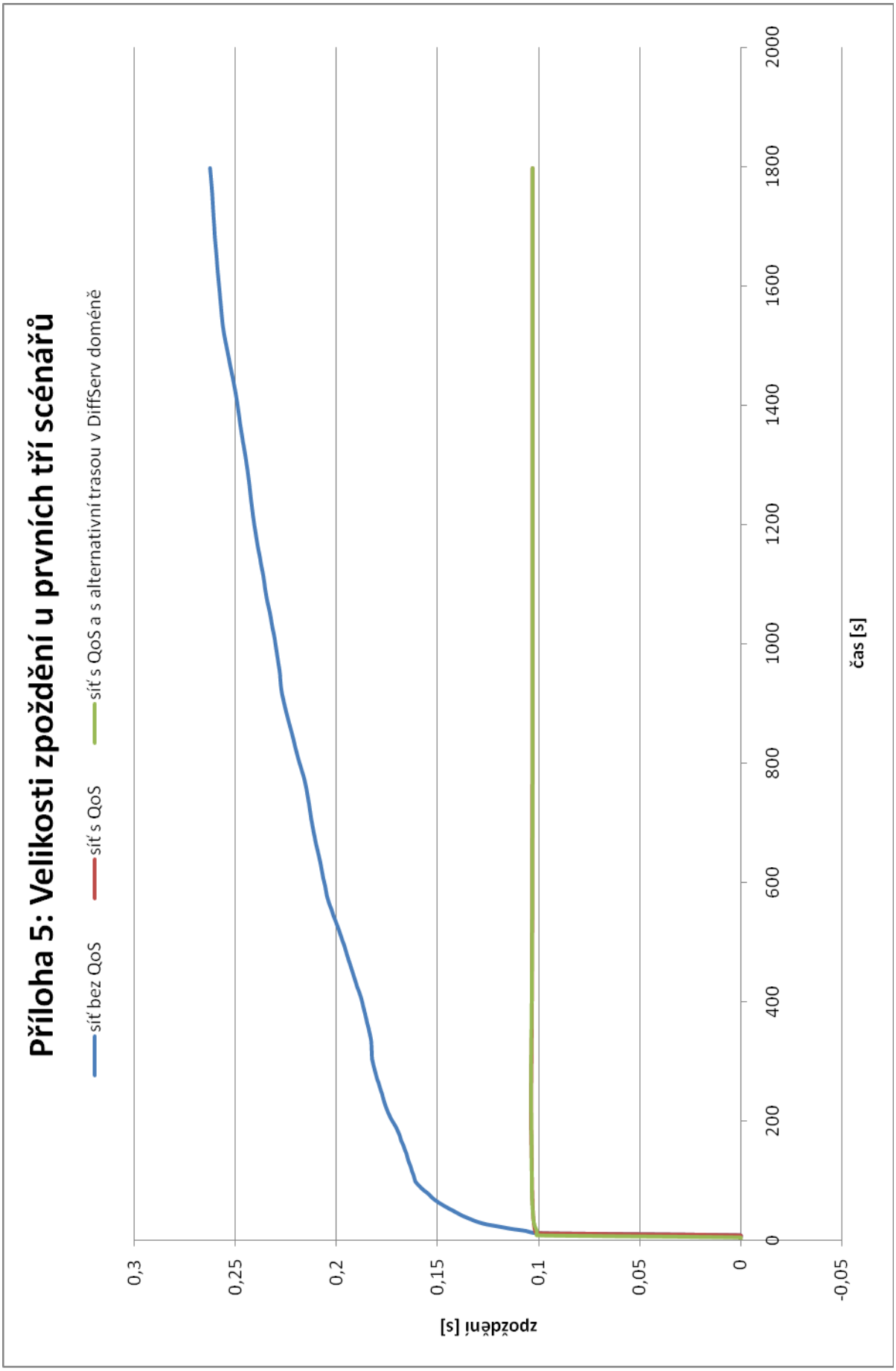
Tab.: Hraniční router Edge_Router_2

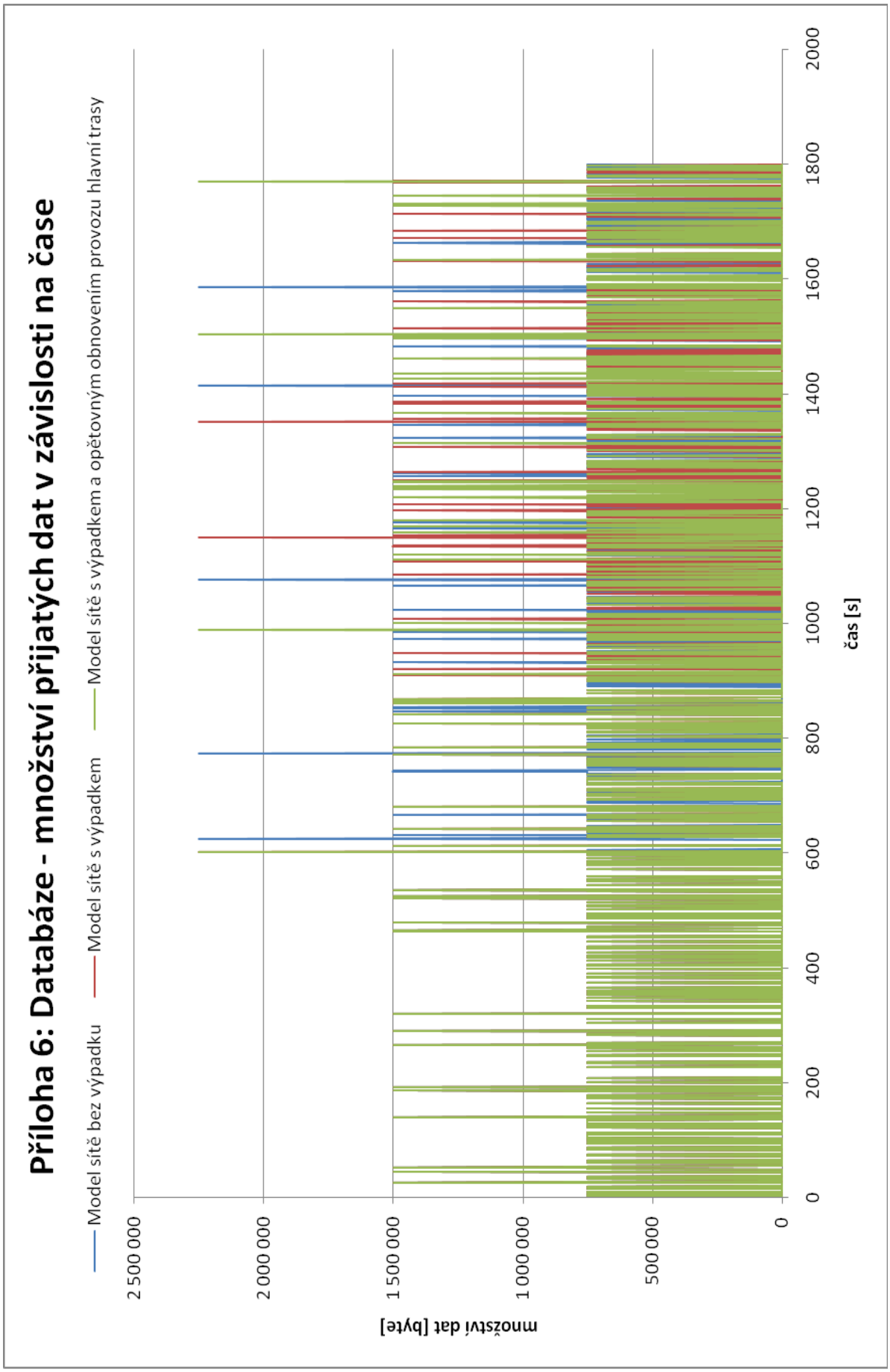
Destination	Source Protocol	Route Preference	Metric	Next Hop Address	Next Hop Node	Outgoing Interface	Insertion Time (secs)
192.0.1.0/24	OSPF 1	110	35	192.0.8.1	Logical Network.Core_Router_4	IF2	62.931
192.0.2.0/24	OSPF 1	110	40	192.0.4.2	Logical Network.Core_Router_2	IF1	62.931
192.0.3.0/24	OSPF 1	110	16	192.0.8.1	Logical Network.Core_Router_4	IF2	62.931
192.0.4.0/24	Direct	0	0	192.0.4.1	Logical Network.Edge_Router_2	IF1	0.000
192.0.5.0/24	Direct	0	0	192.0.5.1	Logical Network.Edge_Router_2	IF0	0.000
192.0.6.0/24	OSPF 1	110	15	192.0.8.1	Logical Network.Core_Router_4	IF2	62.931
192.0.7.0/24	OSPF 1	110	10	192.0.8.1	Logical Network.Core_Router_4	IF2	62.931
192.0.8.0/24	Direct	0	0	192.0.8.2	Logical Network.Edge_Router_2	IF2	0.000
Gateway of last resort is	not set						

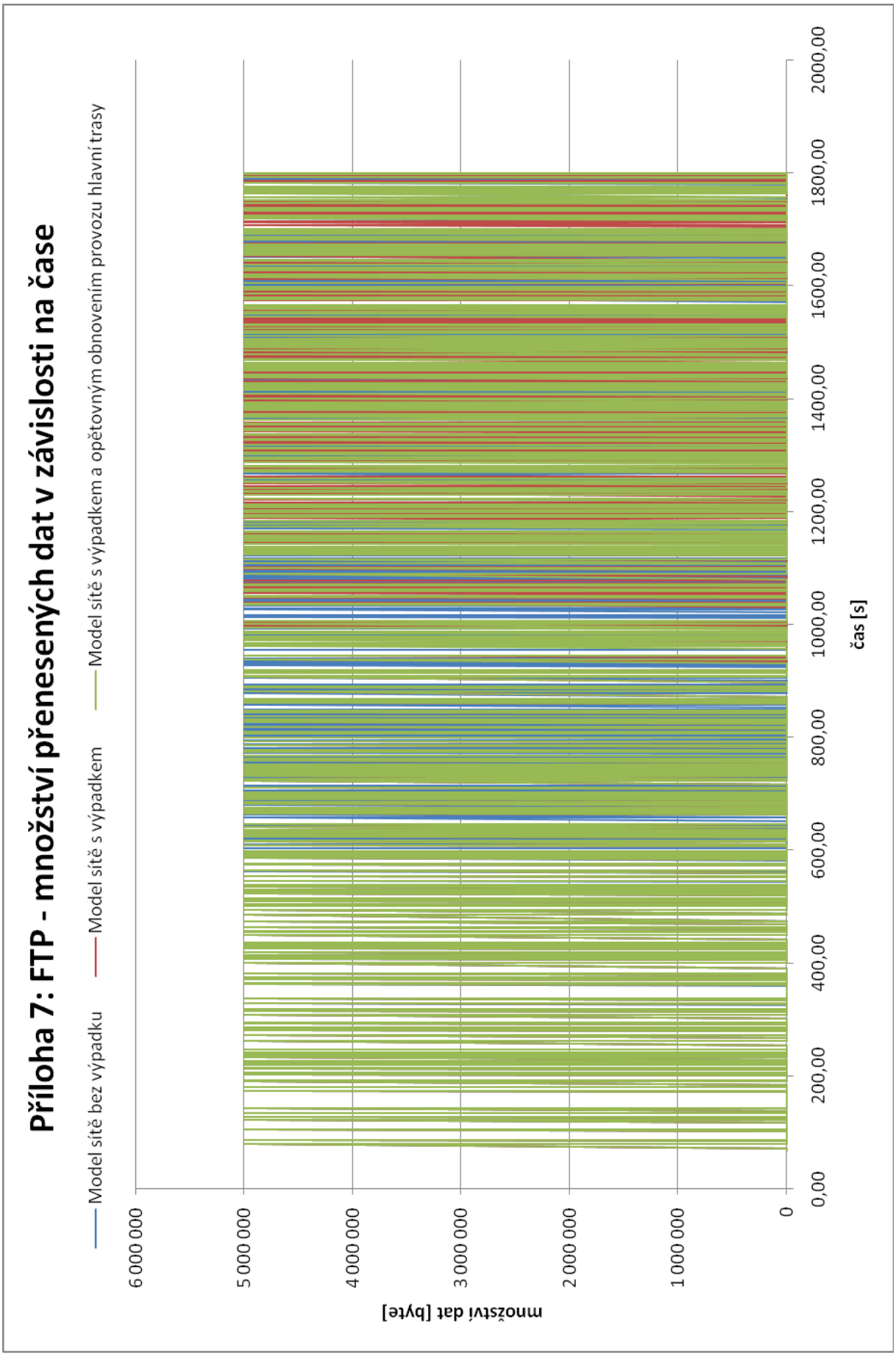

Příloha 2: Počet zahozených paketů v závislosti na čase
Model sítě bez výpadku
Model sítě s výpadkem
Model sítě s výpadkem a opětovným obnovením provozu hlavní trasy
Počet paketů [-]
160
140
120
100
80
60
40
20
0
0
200
400
600
800
1000
1200
1400
1600
1800
2000
čas [s]

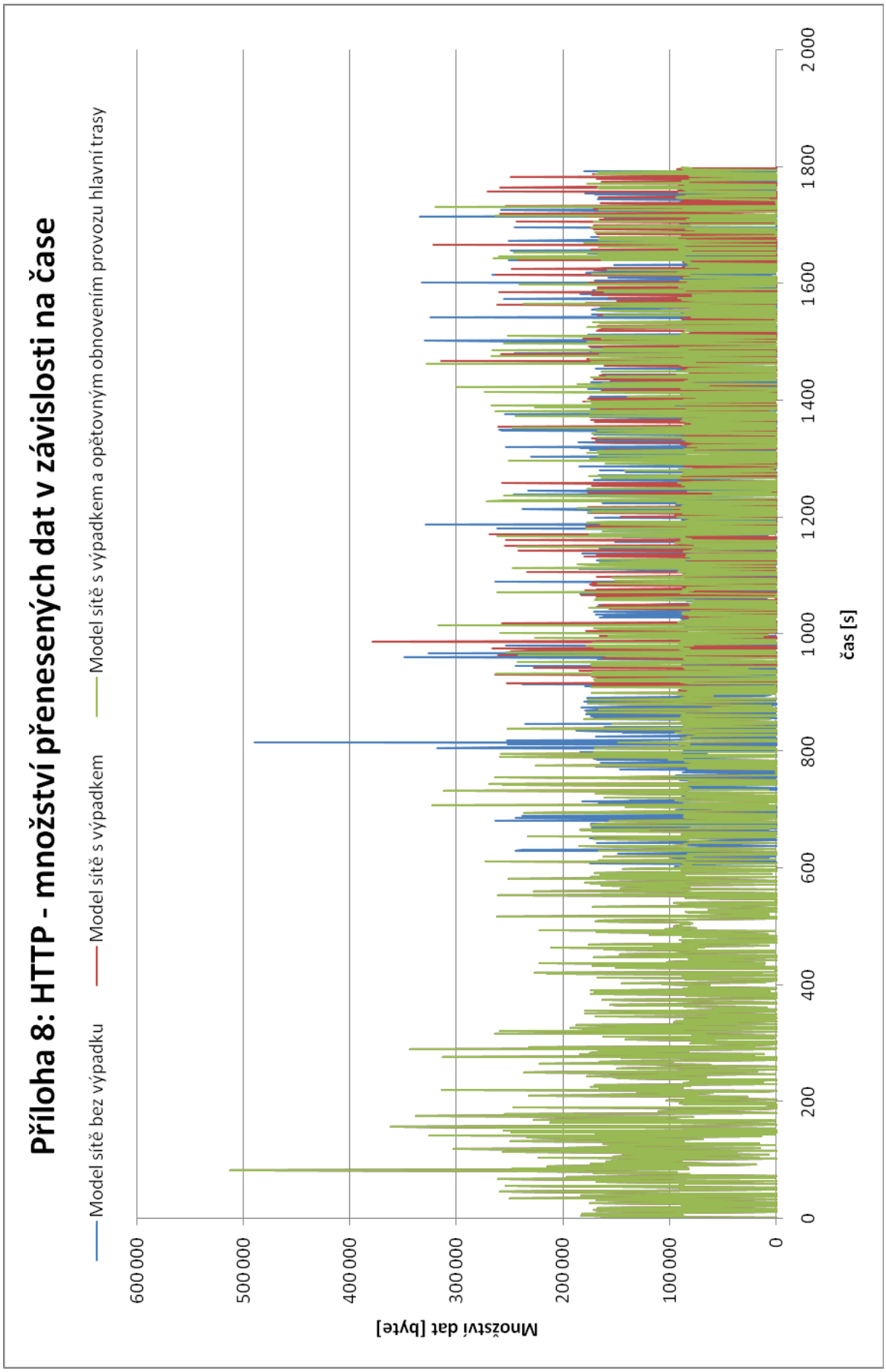

Příloha 3: Porovnání velikosti jitteru u prvních tří scénářů
síť bez QoS
síť s QoS
síť s QoS a s alternativní trasou v DiffServ doméně
jitter [-]
0,00012
0,0001
0,00008
0,00006
0,00004
0,00002
0
-0,00002
0
200
400
600
800
1000
1200
1400
1600
1800
2000
čas [s]


Příloha 4: Velikost jitteru na klientské stanici u prvních tří scénářů
síť bez QoS
síť s QoS
síť s QoS a s alternativní trasou v DiffServ doméně
jitter [-]
0,00004
0,000035
0,00003
0,000025
0,00002
0,000015
0,00001
0,000005
0
-0,000005
-0,00001
0
200
400
600
800
1000
1200
1400
1600
1800
2000
čas [s]


Příloha 5: Velikosti zpoždění u prvních tří scénářů
síť bez QoS
síť s QoS
síť s QoS a s alternativní trasou v DiffServ doméně
zpoždění [s]
0,3
0,25
0,2
0,15
0,1
0,05
0
-0,05
0
200
400
600
800
1000
1200
1400
1600
1800
2000
čas [s]


Příloha 6: Databáze - množství přijatých dat v závislosti na čase
Model sítě bez výpadku
Model sítě s výpadkem
Model sítě s výpadkem a opětovným obnovením provozu hlavní trasy
množství dat [byte]
2 500 000
2 000 000
1 500 000
1 000 000
500 000
0
0
200
400
600
800
1000
1200
1400
1600
1800
2000
čas [s]


Příloha 7: FTP - množství přenesených dat v závislosti na čase
Model sítě bez výpadku
Model sítě s výpadkem
Model sítě s výpadkem a opětovným obnovením provozu hlavní trasy
množství dat [byte]
6 000 000
5 000 000
4 000 000
3 000 000
2 000 000
1 000 000
0
0,00
200,00
400,00
600,00
800,00
1000,00
1200,00
1400,00
1600,00
1800,00
2000,00
čas [s]


Příloha 8: HTTP - množství přenesených dat v závislosti na čase
Model sítě bez výpadku
Model sítě s výpadkem
Model sítě s výpadkem a opětovným obnovením provozu hlavní trasy
Množství dat [byte]
600 000
500 000
400 000
300 000
200 000
100 000
0
0
200
400
600
800
1 000
1 200
1 400
1 600
1 800
2 000
čas [s]

POPIS STRUKTURY PŘILOŽENÉHO DVD

Diplomova_prace

\DSDomain	- složka s projektem v Opnet Modeleru
\Grafy a tabulky	- grafy a zdrojová data grafů použitých v práci
\diplomova_prace.pdf	- elektronický text diplomové práce