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

MOŽNOSTI SIMULACE A OPTIMALIZACE ALGORITMŮ PRO ZAJIŠTĚNÍ QOS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

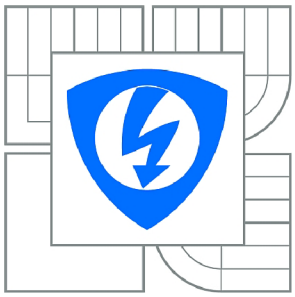
AUTOR PRÁCE
AUTHOR

ONDŘEJ CHARVÁT

BRNO 2015

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

MOŽNOSTI SIMULACE A OPTIMALIZACE ALGORITMŮ PRO ZAJIŠTĚNÍ QOS

POSSIBILITIES OF QOS ALGORITHMS SIMULATION AND OPTIMIZATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

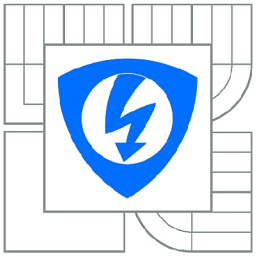
AUTOR PRÁCE — ONDŘEJ CHARVÁT

AUTHOR

VEDOUCÍ PRÁCE — Ing. ONDŘEJ KRAJSA, Ph.D.

SUPERVISOR

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	Ondřej Charvát	***ID:***	147430
Ročník:	3	***Akademický rok:***	2014/2015

NÁZEV TÉMATU:

Možnosti simulace a optimalizace algoritmů pro zajištění QoS

POKYNY PRO VYPRACOVÁNÍ:

S využitím prostředí MATLAB a tollboxu SimEvents navrhněte a realizijte parametrizovatelný systém s grafickým rozhraním zaměřený na problematiku optimalizace zajištění kvality služeb v Ethernetu.

DOPORUČENÁ LITERATURA:

[1] LE BOUDEC, Jean-Yves a Patrick THIRAN. Network calculus: a theory of deterministic queuing systems for the Internet. Berlin: Springer, c2001, xix, 274 s. ISBN 3-540-42184-x.
[2] GIAMBENE, Giovanni. Queuing theory and telecommunications: networks and applications. New York: Springer, c2005, xvii, 585 s. ISBN 0-387-24065-9.
[3] ATTAWAY, Stormy. MATLAB: a practical introduction to programming and problem solving. 2nd ed. Waltham: Butterworth-Heinemann, c2012, xx, 518 p. ISBN 978-0-12-385081-2.
[4] HANSELMAN, Duane C a Bruce LITTLEFIELD. Mastering MATLAB. 1st ed. Upper Saddle River: Pearson, c2012, xiv, 843 p. ISBN 9780136013303.
[5] FRIKHA, Mounir. Ad hoc networks: routing, QoS and optimization. London: ISTE, 2011, x, 266 s. ISBN 978-1-84821-227-5.
[6] BALAKRISHNAN, Ram. Advanced QoS for multi-service IP/MPLS networks. Indianapolis: Wiley Publishing, 2008, 432 s. ISBN 978-0-470-29369-0.

Termín zadání:	9.2.2015	***Termín odevzdání:***	2.6.2015

Vedoucí práce: Ing. Ondřej Krajsa, Ph.D.
Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.
Předseda oborové rady

ABSTRAKT

Tato bakalářská práce se zabývá problematikou řízení datových toků v konvergovaných sítích, které je implementováno za účelem poskytování telekomunikačních služeb v žádané kvalitě. Teoretická část práce nejprve poskytuje přehled měřitelných parametrů kvality služeb a uvádí faktory, kterými jsou ovlivněny. Dále rozebírá různé úrovně poskytovaných služeb a nakonec techniky používané pro zajištění QoS, tak jak jsou implementovány v síťových prostředcích. V praktické části je v simulačním softwaru Simulink zkonstruován model sítě s implementovaným řízením QoS. V této síti probíhá přenos několika datových toků s různými nároky na parametry provozu jako např. telefonie nebo přenos velkých souborů. Prostřednictvím grafického uživatelského rozhraní, vytvořeného v prostředí Matlab, lze měnit nastavení systému QoS v modelu a sledovat vliv na dosažené parametry probíhajících datových toků. Veškeré výsledky simulace jsou taktéž zobrazovány na uživatelském rozhraní. Uživatel má možnost buďto nastavit veškeré parametry ručně, nebo nechat program optimalizovat nastavení podle zadaného kritéria, např. ztrátovost paketů. Model tak demonstruje jaký vliv mají změny konfigurace QoS na kvalitu přenášených služeb. Celý model je zkonstruován tak, aby ho v případě dalšího využití bylo možné jednoduše překonfigurovat. Lze například přidat další datové toky nebo model pomocí předdefinovaných stavebních prvků rozšířit do rozlehlejší síťě.

KLÍČOVÁ SLOVA

QoS, IntServ, DiffServ, zpoždění, prioritní, best effort, DSCP

ABSTRACT

The bachelor's thesis deals with problematics of data transmission control in converged networks which is implemented there to offer telecommunication services in desired quality. Theoretical part of the thesis provides an overview of measurable parameters and lists factors by which they are affected. It discusses different levels of services offered and finally techniques used for it's provisioning as they are implemented in network appliances. In practical part a model of network with QoS implementation is constructed in simulation software Simulink. There are several traffic flows running in the simulated network which has different demands on transfer quality such as telephony or bulky file transfer. There is graphical user interface programmed in Matlab enviroment by which it is possible to alter QoS settings in the model and observe effect on achieved parameters of running data transfers. All simulation results are also displayed on the user interface. User may set all parameters manualy or let the software optimize settings according to prefered criterion, i.e. packet loss. In such way model demonstrates how changes in QoS configuratinon affects quality of services being transferred. The whole model is constructed in a way to be easily reconfigured in case of future use. It is possible for expample to add more traffic or to extend the model to larger network by using predefined blocks.

KEYWORDS

QoS, IntServ, DiffServ, delay, priority, best effort, DSCP

CHARVÁT, Ondřej *Možnosti simulace a optimalizace algoritmů pro zajištění QoS*: bakalářská práce. VUT Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 57 s. Vedoucí práce byl Ing. Ondřej Krajsa, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Možnosti simulace a optimalizace algoritmů pro zajištění QoS“ jsem vypracoval samostatně pod vedením vedoucího bakalářské 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 bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

VUT Brno ....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Ondřeji Krajsovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

VUT Brno ....

(podpis autora)

OBSAH

SEZNAM OBRÁZKŮ

ÚVOD

Na samém počátku éry počítačových sítí nebylo řízení provozu ve smyslu upřednostňování nebo omezování některých datových toků vůbec potřeba. Přenášených služeb nebylo mnoho, byly všeobecně definovány a pro jejich chod existovala dostatečná kapacita. S rozšiřováním sítí do komerčního sektoru, do firem a domácností začala narůstat rozlehlost sítě, množství přenášených dat a hlavně se také rozšiřovalo spektrum služeb využívajících stejnou síť. Dnešnímu trendu, kdy již nejsou paketově orientované počítačové sítě využívány zdaleka jednoúčelově, ale je do nich kombinováno více různých druhů telekomunikačních služeb jako přenos hlasu, videa, prohlížení internetových stránek, přenos velkých souborů a podobně, se říká konvergence. Jelikož přenášených služeb je mnoho a jejich nároky na parametry přenosu jsou různé, je potřeba s dostupnou kapacitou sítě hospodařit tak, aby tyto služby mohly být bez problémů provozovány. Pojem kvalita služeb představuje vhodně definované a kontrolované chování systému v souladu s kvantitativně měřitelnými parametry.[2].

Cílem této práce je rozebrat problematikou zajištění kvality služeb v sítích využívajících protokolovou sadu TCP/IP a linkovou technologii ethernet. Na základě teoretických znalostí bude zkonstruován systém pro provádění simulací, zaměřených na optimalizaci QoS systému.

V teoretické části jsou popsány zejména čtyři základní parametry kvality služeb – propustnost, ztrátovost, zpoždění a kolísání zpoždění. U každého z nich je uvedeno jaké faktory ho určují a jak jej lze příznivě ovlivňovat. Je také zmíněno několik dalších parametrů jako dostupnost služby nebo kvalita prožitku. Dále jsou podrobně rozebrány praktiky používané pro zajištění QoS a jejich kombinace do dvou ucelených architektur diffserv a intserv.

V praktické části je pak s využitím Matlabu a simulačního prostředí Simulink sestrojen model sítě s několika koncovými stanicemi a dvěma směrovači. V tomto systému je implementována QoS architektura typu diffserv, pomocí níž je řízen provoz nadefinovaných uživatelských služeb celkem čtyř typů. Software obsahuje grafické uživatelské rozhraní, pomocí nějž může uživatel snadno měnit nastavení QoS modulů ve směrovačích a také zobrazovat výsledky simulace. Těmi je kvantitativní vyjádření dosažených parametrů přenosu pro jednotlivé přenášené služby. Volba parametrů probíhá buďto v manuálním režimu, nebo automaticky na základě zadaného kritéria, kterým může být propustnost, ztrátovost, zpoždění nebo kolísání zpoždění. Po nastavení konfigurace a spuštění simulace je během několika vteřin, v závislosti na zvolené délce simulace, vidět vliv na dosažené výsledky.

1 TEORETICKÁ ČÁST

1.1 Smysl a cíl zavádění QoS

Systémy pro zajištění QoS (Quality of Services) mají, jak už název napovídá, za úkol zajištění spokojenost zákazníka. Stejně jako v jiném odvětví poskytování služeb zákazník zaplatí za nabízené služby, u kterých jsou deklarovány určité parametry. Úkolem poskytovatele služeb (prodejce) je zajistit zákazníkovi poskytování kýžených služeb s minimálně stejnými parametry, jaké byly deklarovány. Jedině tak bude zákazník spokojený.

V dnešním světě telekomunikací je mnoho služeb, jako přenos hlasu, videa, webových stránek a jiných dat, realizováno pomocí společných, tzv. konvergovaných sítí. Jelikož každá z těchto služeb má různé nároky na přenosové parametry, musejí být konvergované sítě schopny zajistit každé službě takové parametry, které jí umožní bezproblémovou funkci ke spokojenosti jejich uživatelů – zákazníků.

Konkrétní kvalitativní parametry pro poskytované služby jsou mezi dodavatelem a zákazníkem dohodnuty prostřednictvím tzv. SLA (Service Level Agreement). SLA představují smluvní závazek poskytovatele, ve kterém jsou definovány poskytované služby a jejich klíčové parametry. Cílem řízení QoS je právě zajištění dohodnutých parametrů poskytovaných služeb.

1.2 Parametry služeb

Základní technické parametry kvality služeb jsou čtyři:

1.2.1 Propustnost (Thoroughput)

Propustnost je definována jako maximální dlouhodobá zátěž přenosové linky, která je určena počtem přenesených datových jednotek za časovou jednotku (např. pakety za vteřinu). Je dána především těmito faktory:

- rychlostní kapacitou přenovsového média (typ přenosového prostředí, jeho vlastnosti, vlastnosti jeho rozhraní, rušení,...)
- použitou modulací a kódováním na fyzické vrstvě
- objemem režijní komunikace na druhé a třetí vrstvě (záhlaví a zápatí rámců a paketů)
- velikostí datových jednotek třetí vrstvy (se vzrůstající velikostí paketu vzrůstá poměr uživatelských dat a celkově přenášených dat včetně těch režijních)
- využitelnou propustností dle dané třídy QoS a vytížeností sítě

Propustnost může být různá pro různé třídy přenosu - viz. kapitola 1.3. Největší nároky na propustnost mají videokonferenční služby, služby pro přenos streamovaného videa a celkově přenos velkých objemů dat.[1][2][3]

1.2.2 Ztrátovost (Packet loss)

Ztrátovost představuje množství paketů, které byly na cestě od zdroje k cíli zahozeny. Definuje se poměrem ztracených paketů a celkovým počtem odeslaných paketů.

K zahození paketu může dojít při zahlcení sítě, kdy se zaplní fronty směrovacích prvků a ty poté musí rozhodnout o tom, které pakety doručí a které ne v závislosti na typu služby, kterou paket nese, a dohodnutých podmínkách. Dalším důvodem může být bitová chybovost způsobená interferencemi nebo zvýšeným útlumem na fyzické vrstvě. Tyto bitové chyby jsou pak detekovány vyššími vrstvami, které pokud je nejsou schopné opravit, rozhodnou o zahození celého paketu. Ke ztrátě paketu může dojít také výpadkem některého síťového prvku, kdy není možné vyslané datové jednotky doručovat cíli, dokud není nalezena alternativní cesta k cíli. Při ztrátě některého paketu pak záleží na protokolu transportní vrstvy, zda-li zažádá o opětovné zaslání daného paketu (TCP) nebo ne (UDP). V případně služeb provozovaných v reálném čase, jako je například telefonní hovor, by opětovné zaslání paketu nemělo pro uživatele význam. U těchto služeb je určitá ztrátovost tolerována. Naopak například při přenosu souborů by mělo nedoručení některé části dat fatální následky.[2][3]

1.2.3 Zpoždění(Delay)

Zpoždění je definováno jako doba, kterou trvá přenos paketu od zdroje k cíli. Při každém přenosu dochází ke zpoždění způsobeným několika faktory:[2]

- **Zpoždění při přenosu**, které je dané konečnou rychlostí průchodu každého signálového prvku přenosovým médiem. Závisí na druhu přenosového média a linkové vzdálenosti od zdroje k cíli. I když zpoždění při přenosu mezi dvěma přímo propojenými uzly sítě je konstantní, z pohledu přenosu od zdroje k cíli už tomu tak není. Rychlost průchodu signálu přenosovým médiem jistě ovlivnit neumíme (jinak než změnou média) ovšem co ovlivnit můžeme je linková vzdálenost od zdroje k cíli. Slůvko "linková"upozorňuje na fakt, že i když je geografická vzdálenost zdroje a cíle konstantní (vyjma mobilních zařízení), vzdálenost cesty přenosu dat mezi nimi konstantní není a to díky existenci alternativních linkových propojení. Jediný způsob jak ovlivnit zpoždění při přenosu je tedy řídit směrování za cílem co nejvíce zkrátit linkovou vzdálenost zdroje a cíle. Běžné hodnoty zpoždění při přenosu se pohybují v řádech jednotek milisekund na 1000 km.[3]

- **Zpoždění při přepínání**, které vzniká ve směrovačích. Je to doba od příjmu paketu po jeho zařazení do výstupní fronty směrovače. Je-li přepínání realizováno softwarově, pak se toto zpoždění pohybuje okolo 2-3 ms na jeden paket. V případě páteřních směrovačů, kde je většinou přepínání implementováno hardwarově, je to dokonce jen 10-20 μs na paket, čímž lze tento druh zpoždění považovat za téměř zanedbatelný.[3]
- **Zpoždění čekáním ve frontě** je definované časem od zařazení paketu do výstupní fronty směrovače po započetí jeho odesílání na výstupní rozhraní. Toto zpoždění závisí na algoritmu použitém pro plánování paketů a na vytížení výstupní fronty, což dále závisí na kapacitě této fronty a vytížení linky. Je tedy značně proměnné.[3]
- **Zpoždění paketizací a serializací** je důsledkem nutnosti vlastního zapouzdřování dat do paketů a rámců a jejich následného kódování na výstupní fyzické médium, kde je paket odesílán sériově bit po bitu. Jeho velikost závisí na rychlosti linky a velikosti paketu. Pro paket o velikosti 1500 bitů je při rychlosti linky 10 Mbps serializační zpoždění asi 1 ms. Není jej možné snížit jinak než navýšením rychlosti linky.[3]

Zpoždění je kritické pro aplikace pracující v reálném čase, jako hovory a videohovory, hraní online her a podobně.

1.2.4 Kolísání zpoždění (Jitter)

Kolísání zpoždění charakterizuje variabilitu zpoždění v síti při přenosu. Výklad tohoto pojmu není vždy jednoznačný, protože kolísání může být například vztaženo k průměrnému či minimálnímu zpoždění. Podle RFC3393 je kolísání zpoždění standardizováno pod pojmem IP Packet Delay Variation jako rozdíl ve zpoždění dvou následujících paketů. Je způsobené proměnlivostí jednotlivých druhů zpoždění popsaných v kapitole 1.2.3 výše. Zpoždění při přenosu se změní, změní-li se topologie sítě vlivem např. výpadkem některé části spojení nebo nalezením rychlejší nebo levnější cesty pro pakety dané služby. Ze stejného důvodu se pak mění i serializační zpoždění, protože nové cesty mohou mít různou rychlost od těch původních. Zpoždění přepínáním může být různě dlouhé díky odlišnosti v náročnosti na zpracování jednotlivých paketů. Nakonec proměnné zpoždění čekáním ve frontě, které je způsobeno měnící se délkou front na výstupech směrovačů a má na celkové kolísání zpoždění největší dopad. Proměnlivost zpoždění představuje největší problém pro hlasové služby, kde je nutné vysílat a přijímat každých 20 ms jeden paket. Použitím zpožďovacího bufferu je možné kolísání zpoždění vyrovnat na konstantní hodnotu.[2][3]

1.2.5 Další parametry kvality služeb

Mezi další parametry služeb patří:

- **Zachovávání pořadí přenášených jednotek** – Díky nespojově orientovanému přenosu, kdy každý paket může být k příjemci přepravován jinou cestou, můžou datové jednotky dorazit do cíle v jiném pořadí než byly odeslány. Z tohoto důvodu jsou datové jednotky číslovány pořadovým číslem a pokud příjemce obdrží paket s nižším pořadovým číslem než měl předešlý paket, pak musí správné pořadí paketů obnovit. Měřítkem je v tomto případě poměr paketů, které přišly v nesprávném pořadí, k celkovému počtu doručených paketů. Snížení tohoto poměru lze dosáhnout implementací takových algoritmů QoS a algoritmů pro řízení zátěže v síti, které jsou orientovány na soustředění jednoho toku paketů stejnou cestou.[3]

- **Dostupnost služby** – Může být definována nezávisle na dostupnosti sítě, kdy její maximální hodnota je omezená dostupností sítě nebo může reflektovat dostupnost služby pouze v době, kdy je síť dostupná. V obou případech musí být hodnocena nezávisle na ostatních parametrech SLA jako zpoždění, kolísání zpoždění a ztrátovost paketů. Dostupnost služby je definována je jako podíl času, kdy je služba dostupná. Podobně je definována i dostupnost sítě, která reflektuje podíl času, kdy je vstupní a výstupní bod sítě v rámci jednoho SLA obousměrně propojen. Dostupnost sítě mohou negativně ovlivnit jak plánované odstávky kvůli pravidelné údržbě sítě nebo přestavbám, tak i neočekávané výpadky sítě.[3]

- **Kvalita prožitku** – Anglicky "Quality of Experience"(QoE) definuje speciální metriku reflektující kvalitu služby poskytovanou koncovou aplikací tak, jak ji vnímá koncový uživatel. Nezaměřuje se tedy přímo na jednotlivé parametry přenosu, ale na jejich celkový vliv na prožitek koncového uživatele služby. Je definována speciálně pro aplikace pracující v reálném čase jako přenos hlasu, videa a hraní her online. Pro první dvě zmíněné služby je QoE dána kombinací kvality enkodéru na vysílací straně, kvality služby přenášející data IP sítí (kombinace propustnosti, zpoždění,...) a kvality dekodéru na přijímací straně. Kvalita prožitku může být měřena buďto subjektivně nebo objektivně. V případě přenosu hlasu se pro subjektivní měření kvality prožitku používá metoda "Mean Opinion Score"(MOS), která poskytuje číselné vyjádření kvality hlasu na přijímací straně. Je to oficiálně definovaná metoda ITU, dle které je sada standardizovaných vět zaznamenaná na vysílací straně a přenesena do přijímače, kde je následně reprodukována několika lidským posluchačům,

kteří následně subjektivně hodnotí kvalitu reprodukovaných vět na stupnici 1-5. Objektivní metody využívají pro vyhodnocení kvality služby jak vysílaný tak přijímaný audiosignál a jejich cílem je v podstatě předpovědět jaké by bylo subjektivní hodnocení MOS. Analogické techniky jsou využívány i pro subjektivní a objektivní měření kvality videa.[3]

1.3 Úrovně poskytovaných služeb

1.3.1 Služby best effort

Poskytování služeb typu best effort se prakticky neřídí žádnými pravidly QoS. Síť se snaží doručit pakety co nejefektivněji od zdroje k cíli, ale služba neposkytuje žádné záruky propustnosti, zpoždění, kolísání zpoždění ani ztrátovosti paketů. Její výhodou je snadná implementace, ale při zvyšující se zátěži sítě již není pro většinu služeb použitelná, protože nerozlišuje různé kvalitativní požadavky různých typů datových proudů. Například je-li síť zahlcena, jsou pakety při zaplnění front bez rozdílu zahazovány systémem tail drop (viz dále). Byl to původní koncept poskytování služeb přes internet.[3]

1.3.2 Služby s řízením zátěže

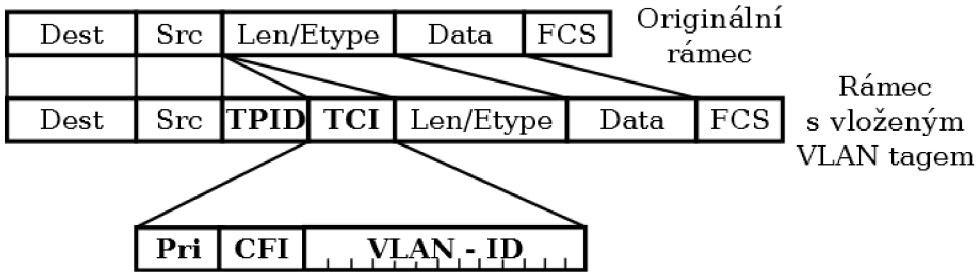
Tato úroveň služby zajišťuje podobné parametry přenosu jako best effort při nízké zátěži sítě s tím rozdílem, že tyto parametry udrží i při vysoké zatíženosti sítě. Je vhodná pro aplikace, které tolerují určité procento ztrátovosti paketů a určité zpoždění, aniž by to výrazně omezilo jejich funkčnost.[6]

1.3.3 Garantované služby

Zaručují 100% doručení veškerých datových jednotek (nulová ztrátvost), pevné maximální zpoždění a pevnou minimální propustnost nezávisle na zatížení sítě. Kapacita sítě pro tyto služby tedy musí být pevně rezervována. Je určena pro náročné aplikace pracující v reálném čase, které netolerují nestabilitu přenosu.[6]

1.4 Architektury zajištění QoS

Z pohledu rozdělení síťové architektury na jednotlivé vrstvy, ať už dle modelu ISO/OSI nebo TCP/IP, bylo potřeba se zamyslet, do které vrstvy nebo vrstev bude nejvhodnější implementovat techniky pro zajištění QoS. Z pohledu ovlivňování klíčových parametrů služeb se zde nabízí síťová a linková vrstva. Jelikož působnost



Obr. 1.1: Rozšíření ethernetového rámce o VLAN tag[4]

linkové vrstvy je omezena na komunikaci bezprostředně si sousedících entit, je z pohledu zajištění end-to-end kvality služeb vhodnější realizovat algoritmy QoS na vrstvě síťové, která řeší komunikaci koncových uživatelů služby skrz celou cestu sítí. Avšak dodržení SLA by nebylo možné bez podpory linkové vrstvy, která poskytuje síťové vrstvě právě komunikaci sousedících si uzlů.[3]

1.4.1 Podpora QoS na linkové vrstvě

V původní specifikaci ethernetu IEEE 802.3 nebyla podpora QoS vůbec zahrnuta. Až s nově zavedenou podporou VLAN sítí, která byla standardizována ve specifikaci IEEE 802.1Q, je možné přiřadit rámcům různou prioritu. Do ethernetového rámce bylo vloženo 32bitové pole, tzv. VLAN tag nebo také 802.1Q halvička, jak ukazuje obr.1.1[4]

První dva bajty tohoto pole jsou označeny jako TPID (Tagged Protocol Identifier) a slouží k identifikaci typu rámce. Pokud je hodnota TPID rovna 0x8100, pak se jedná o tagovaný rámec dle 802.1Q. Jelikož je TPID uložen hned za zdrojovou hradwarovou adresou, tedy na stejném místě, kde je v obyčejném (neotagovaném) rámci pole "EtherType", je v případě jiné hodnoty než 0x8100 správně rozpoznáno, že se jedná o obyčejný rámec a daná hodnota je zpracována jako "EtherType". Další část VLAN tagu je TCI (Tagged Control Information) a je rozdělena do tří částí:

- VLAN-ID, označující do které VLAN rámec patří
- jednobitové pole CFI (Canonical Format Indicator), které dříve označovalo formát adresy z důvodu kompatibility ethernetu s technologií token ring. Dnes již toto není využíváno, a proto bylo pole CFI ve specifikaci 802.1Q-2011 změněno na DEI (Drop Eligible Indicator) a označuje, zda-li je možné v případě přetížení sítě rámec zahodit.
- tříbitové pole PCP (Priority Code Point), které definuje 8 prioritních tříd viz tab.1.2 Kombinací polí DEI a PCP lze tedy řídit kvalitu služeb na linkové

Priorita	Zkratka	Typ přenosu
1	BK	Pozadí
0 (základní)	BE	Best effort (nejlepší snaha)
2	EE	Excellent effort (prvotřídní snaha)
3	CA	Kritická aplikace
4	VI	„Video“ >100 ms zpoždění a kolísání zpoždění
5	VO	„Hlas“ >10 ms zpoždění a kolísání zpoždění
6	IC	Internetové řízení
7	NC	Síťové řízení

Obr. 1.2: Prioritní třídy QoS jak jsou definované v linkové vrstvě ethernetu[5]

Počet front	Definovaný typ služby							
1	BE							
2	VO				BE			
3	NC		VO		BE			
4	NC		VO		CA		BE	
5	NC	IC	VO		CA		BE	
6	NC	IC	VO		CA		BE	BK
7	NC	IC	VO		CA	EE	BE	BK
8	NC	IC	VO	VI	CA	EE	BE	BK

Obr. 1.3: Definované prioritní třídy při nižším počtu front[5]

úrovni.[5]

Ne všechny tyto třídy jsou vždy při řízení QoS využity. Skutečný počet využitých tříd závisí na počtu front, které jsou na příslušném síťovém prvku k dispozici pro rozdělení agregovaného datového toku. Které třídy jsou použity při určitém počtu front ukazuje tab.1.3

Jak se ale dostane do záhlaví rámce informace o požadované třídě služeb, když řízení QoS probíhá na síťové vrstvě? Odpověď je prostá: Mapováním informací o prioritě z odpovídajících polí záhlaví síťové vrstvy do VLAN tagu v záhlaví linkové vrstvy. Podobně je tomu i při využití MPLS. Pravidla pro toto mapování však nejsou standardizována v žádné normě a jsou proto v rukou poskytovatelů teleko-

munikačních služeb. Z tohoto důvodu se může konkrétní způsob mapování priorit mezi vrstvami během cesty sítí lišit. [5]

1.4.2 Nástroje používané při řízení toků

Aby byly dodrženy parametry služeb dle SLA, chovají se mechanismy pro zajištění QoS jako nadřazené prvky nad mechanismy směrování. Směrování jako takové probíhá pomocí k tomu určených protokolů, ale architektury, které zajišťují kvalitu služeb, do tohoto procesu zasahují ve smyslu zajištění požadovaných parametrů. K tomuto používají řadu nástrojů, které si nyní popíšeme.[3]

Klasifikace paketů (Classification)

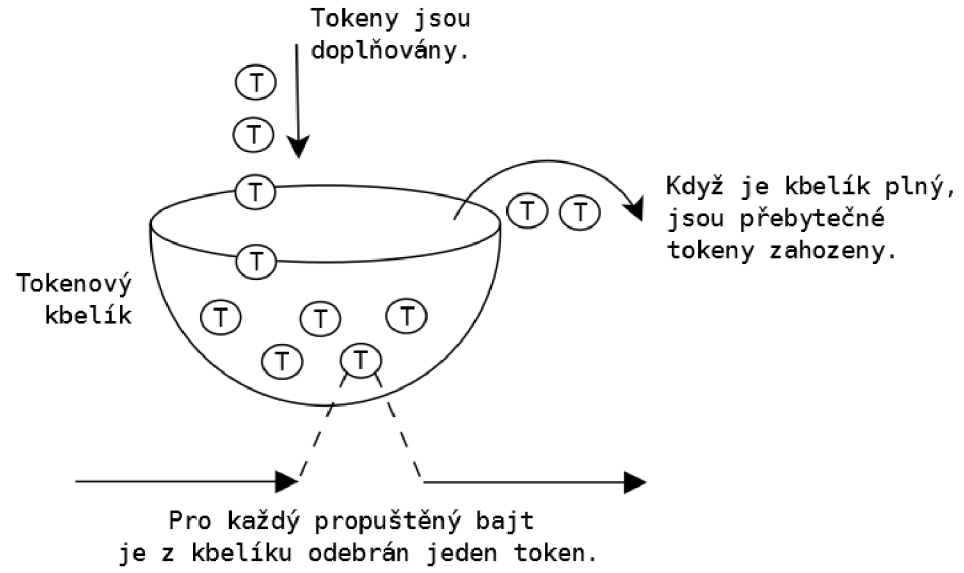
Je to proces rozlišování jednotlivých datových toků a jejich seskupování do určitých tříd, se kterými bude následně zacházeno podle jednotných pravidel. Známe několik druhů klasifikace paketů, které jsou dle potřeby různě kombinovány:

- Jednoduchá klasifikace – k rozlišování paketů využívá pouze k tomu určená pole jejich záhlaví a nepotřebuje tedy znát žádné další podrobnosti o paketu ani datech v něm přenášených
- Implicitní klasifikace – využívá pouze informací z první a druhé vrstvy jako je adresa rozhraní, na kterém byl rámec obdržen
- Komplexní klasifikace – využívá informací v záhlaví paketu, které nejsou explicitně určeny pro řízení QoS jako adresa a port odesílatele a příjemce, použitý protokol, nebo také např. využívat informace ze záhlaví rámce druhé vrstvy
- Hloubkový průzkum paketu – jsou analyzována vlastní přenášená data v paketu a tyto jsou pak použity pro klasifikaci.
- Stavový průzkum paketu - provádí klasifikaci na základě určitých stavových informací v paketu, pomocí kterých je následně schopen paket přiřadit k již existujícímu (klasifikovanému) toku nebo ho zahodit.

Poslední dvě techniky jsou zvláště užitečné pokud je některé datové toky obtížné identifikovat. Takovým tokem může být např. peer-to-peer přenos, který, jelikož značně zatěžuje síť a bývá tedy poskytovateli služeb často omezován, se může snažit zamaskovat za jiný datový tok.[3]

Značkování (Marking)

Tato technika, známá také jako "obarvování“ paketů, spočívá v nastavování hodnot v polích paketů určených pro řízení QoS, aby následně mohly být pakety jednoduše



Obr. 1.4: Schematické znázornění funkce omezování pomocí "tokenového kbelíku"[7]

identifikovány. Pakety jsou většinou označkovány již na vstupu do sítě pod jedním SLA a dále jsou tyto informace pouze čteny pro jejich klasifikaci.[3]

Měření a omezování (Metering and policing)

Používá se k zamezení překročení stanoveného rychlostního limitu pro učitý datový tok. Schematicky bývá zobrazován jako tzv. token bucket neboli tokenový kbelík - obr.1.4. Do kbelíku jsou v pravidelném intervalu doplňovány tokeny, kde každý token odpovídá jednomu bajtu. Přijde-li v datovém toku paket, je zkontrolováno, zda-li je v kbelíku dostatečné množství tokenů odpovídající velikosti paketu. Pokud ano, je paket označen jako "vyhovující", je zpracován a z kbelíku je odebráno příslušné množství tokenů. Pokud ne, pak je paket označen jako „přesahující limit“ a je zahozen nebo označen např. jako vhodný kandidát pro zahození v případě přetížení sítě. Pokud je kbelík pravidelně dávkovanými tokeny již naplněn, přetékající tokeny jsou zahozeny. Hloubka kbelíku tedy určuje maximální dávku příchozích paketů, která může být obsloužena.[3]

Toto jednoduché schéma je využito i u sofistikovanějších aplikací jako je například "tříbarevné zančení"paketů napodobující pokyny semaforů v dopravě. Jsou definovány dva tokenové kbelíky (C=confor, E=exceed), které jsou oba periodicky plněny tokeny. Pokud příchozí paket může být obsloužen tokeny z kbelíku C, pak je označen jako zelený (vyhovující) a je zpracován. Pokud není v kbelíku C dostatek tokenů, ale je jich dostatek v kbelíku E, je paket označen jako oranžový (nevyhovující) a je přenesen. Pokud není dostatek tokenů ani v jednom z kbelíků, pak je paket

označen červeně (porušující zásady) a je zpravidla zahozen.[3]

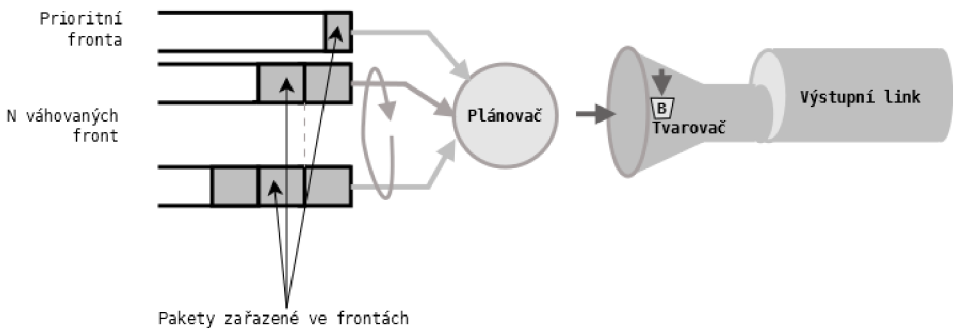
Plánování a řazení do front (Scheduling and Queuing)

Je to proces uplatňování vlastních pravidel řízení datového toku dle SLA. Nemohou-li být všechny pakety při příchodu do síťového uzlu obslouženy ihned (například směřuje-li zde více vstupních datových toků do jednoho výstupního rozhraní), musejí být zařazeny do front, které jsou následně postupně obsluhovány dle kombinace určitých pravidel plánování.[3]

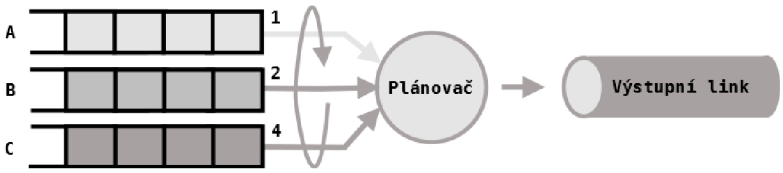
- ***Prioritní plánování*** – Většina dnešních plánovačů podporuje prioritní frontu pro upřednostnění služeb citlivých na zpoždění. Tato fronta je pak obsluhována přednostně, nezávisle na ostatních pravidlech, kdy může být její upřednostnění uplatněno několika způsoby:
  - **Preemptivně**, kdy je prioritní fronta obsloužena jakmile se stane aktivní. Zde ještě rozeznáváme dva druhy preemptivního zpracování:
    - **Na paketové úrovni** – zpracování aktuálního paketu v jiné než prioritní frontě je přerušeno = nejrychlejší odezva, ale neprioritní paket musí být odeslán znovu, což snižuje celkovou efektivitu sítě
    - **Na množstevní úrovni** – zpracovávání aktuální neprioritní fronty je přerušeno, ale odesílání aktuálně zpracovávaného neprioritního paketu je dokončeno. V současnosti je tento systém nejvyužívanější.
  - **Nepreemptivně**, kdy je prioritní fronta zařazena ke zpracování hned za aktuálně zpracovávanou neprioritní frontu.

  Pokud by byla prioritní fronta stále plně vytížena, datové toky v ostatních frontách by nebyly nikdy obslouženy. Z tohoto důvodu bývá prioritní fronta omezována, aby v ní provoz nedosáhl celkové výstupní kapacity síťového uzlu a byl zde také prostor pro zpracování neprioritních dat. S tímto musí být samozřejmě počítáno při definování podmínek SLA resp. při návrhu sítě, která má určitým SLA vyhovovat.[3]

- ***Váhové plánování*** – Pokud zrovna není obsluhována prioritní fronta, jsou obsluhovány fronty ostatní, které jsou také různě prioritizovány. Priority jsou frontám přiřazovány podle toho, jak kvalitativně náročné datové toky nebo třídy provozu tyto fronty obsluhují (dle SLA). Plánovací algoritmus pak rozděluje disponibilní propustnost výstupního rozhraní mezi tyto fronty. Exitují tři způsoby váhového plánování:
  - **Weighted Round Robin** (WRR) neboli váhová cyklická obsluha je nejjednodušším způsobem váhového plánování. Vysvětleme jeho funkci



Obr. 1.5: Znázornění plánovače s prioritní frontou[3]



Obr. 1.6: Příklad WRR plánovače se třemi frontami[3]

na příkladu třech front viz obr.1.6, kde fronta A má váhu 1, fronta B váhu 2 a fronta C má váhu 4.
Při každém cyklu plánovač navštíví každou z front a obslouží z ní takový objem dat, který odpovídá její váze. V našem příkladu tedy z fornty A jeden paket, z fronty B dva pakety a z fronty C čtyři. V případě, že jsou všechny pakety ve všech frontách stejně velké, je tímto způsobem každé frontě přiřazen podíl kapacity výstupní linky daný její váhou. Označíme-li si váhy jednotlivých front w_X a jejich přidělenou kapacitu C_X, kde X je označení fronty, a celkovou kapacitu výstupního rozhraní C_{out}, pak můžeme snadno vyjádřit kapacitu přidělenou každé frontě jako:

$$C_X = \frac{w_X}{\sum_{i=1}^{n} w_{Xi}} C_{out} \tag{1.1}$$

V našem případě by při celkové kapacitě např. 512 kbps byla tato kapacita rozdělena mezi jednotlivé fronty následovně:

$$C_A = \frac{1}{1+2+4} 512 = 73\ kbps$$
$$C_B = \frac{2}{1+2+4} 512 = 146\ kbps$$
$$C_C = \frac{4}{1+2+4} 512 = 293\ kbps$$

Pro zefektivnění provozu v případech, kdy je některá z front v okamžiku její obsluhy neaktivní (prázdná), je tato fronta přeskočena a její kapacita je rozdělena ostatním frontám dle vzájemného poměru jejich vah. Plánovače, které tuto techniku mají implementovanou se nazývají „work conserving“ plánovače (plánovače s úsporou práce).[3]
V tomto nejjednodušším případě, kdy je velikost všech paketů stejná, by bylo takto možné zajišťovat přidělení dohodnuté propustnosti dle SLA. Pokud se však velikost paketů liší, nepřiděluje již tato jednoduchá technika dostupnou propustnost odpovídajícím způsobem, protože operuje se všemi frontami tak, jako kdyby obsahovala pakety stejné délky. Částečným řešením je podělit váhu každé fronty průměrnou velikostí jejích paketů:

$$C_X = \frac{w_X s_{avgX}}{\sum_{i=1}^{n} w_{Xi}} C_{out}, \text{ kde } s_{avgX} \text{ je průměrná velikost paketu ve frontě X} \tag{1.2}$$

Tady ovšem možnosti WRR končí a pokud se průměrná velikost paketu jednotlivých toků během provozu mění, jak tomu také v praxi bývá, snižuje se spravedlivost rozdělování propustnosti mezi různé fronty dle SLA. Propracovanější systémy plánování dokáží toto omezení překonat a přiblížit se modelu tzv. zobecněného sdílení procesu (Generalized Process Sharing (GPS)). Tento teoretický model obslouží při každém cyklu jen nekonečně malou část z každé fronty. Tímto je schopen obsloužit v jakkoli krátkém časovém intervalu všechny fronty a vykonává tak ideálně spravedlivé dělení prospustnosti mezi ně. Reálně sice tohoto pochopitleně nelze dosáhnout, ale cílem plánovacích algoritmů je tomuto modelu se co nejvíce přiblížit.[3]

– **Weighted Fair Queuing (**WFQ) neboli váhované spravedlivé řazení do front vypočítává čas, kdy by bylo ukončeno zpracování jednotlivých paketů systémem GPS. Ve stejném pořadí, v jakém jdou za sebou tyto vypočítané časy, jsou následně pakety zpracovávány. V reálné implementaci však samozřejmě nelze vypočítávat pořadí z nekonečně malých částí a je proto využívána jednotka bajt. Pro účely výpočtu pořadí je v plánovači zavedeno tzv. počítadlo kol, které počítá kolik cyklů zpracování bajt po bajtu již plánovač vykonal. Když dorazí paket do zatím prázdné (neaktivní) fronty, je jeho pořadí vypočítáno jako součet počítadla kol a velikosti bajtu vynásobeného váhou příslušné fronty. Dorazí-li paket do již aktivní fronty, je pořadí vypočítáno jako jeho velikost vynásobená váhou fronty, ke které přičteme pořadí paketu s nejvyšším pořadovým

číslem ve frontě. Jelikož takto získají dřívější pořadí pakety ve frontách s nižší váhou, musejí být váhy přiřazovány tak, že nejnižší číslo znamená nejvyšší váhu, tedy opačně než tomu bylo u techniky WRR. Fronta je aktivní, pokud se v ní nachází paket s vyšším pořadovým číslem, než je aktuální hodnota počítadla kol.[3]

Pro jasnější vysvětlení se vraťme k předchozímu příkladu, kde definujeme konstantní velikost paketů v jednotlivých frontách $s_A = 64$ bytů, $s_B = 1500$ bytů a $s_C = 300$ bytů a pořadí plnění jednotlivých front takto: A1, A2, B1, C1, C2, C3. Předpokládejme, že pakety přicházejí rychleji než je plánovač schopen obsloužit první paket. Váhy front přiřadíme inverzně takto: $w_A = 4$, $w_B = 2$, $w_C = 1$. Počítadlo kol označme r a nastavme jeho počáteční stav na 0 a pořadí zpracování paketů označujme $n_X i$.

1. Paket A1 dorazil do neaktivní fronty a jeho pořadové číslo tedy bylo vypočítáno jako $r + s_A * w_A = 0 + 64 * 4 = 256$
2. Paket A2 dorazil a jelikož je již fronta aktivní, jeho pořadové číslo bylo vypočítáno jako $n_A 1 + s_A * w_A = 256 + 64 * 4 = 512$
3. Paket B1 dorazil do neaktivní fronty a jeho pořadové číslo tedy bylo vypočítáno jako $r + s_B * w_B = 0 + 1500 * 2 = 3000$
4. Paket C1 dorazil do neaktivní fronty a jeho pořadové číslo tedy bylo vypočítáno jako $r + s_C * w_C = 0 + 300 * 1 = 300$
5. Paket C2 dorazil do již aktivní fronty a jeho pořadové číslo tedy bylo vypočítáno jako $n_C 1 + s_C * w_C = 300 + 300 * 1 = 600$
6. Paket C3 dorazil do již aktivní fronty a jeho pořadové číslo tedy bylo vypočítáno jako $n_C 2 + s_C * w_C = 600 + 300 * 1 = 900$

Plánovač pak pakety obsluhuje podle vypočítaného pořadového čísla, čímž se pořadí odesílání paketů změní na A1, C1, A2, C2, C3, B1.[3]

– **Deficit Round Robin (**DRR) neboli deficitní cyklická obsluha je obodbná WRR s tím, že řeší problém s různou velikostí paketů počítáním tzv. deficitu pro každou frontu. Množství dat, které má být v každé frontě obslouženo a vypočítává se z váhy dané fronty, se nepočítá na pakety, ale na bajty. Pak je při každém kole obslouženo tolik paketů z dané fronty, kolik se vejde do přiděleného konta bajtů dané fronty. Jelikož v závislosti na aktuální velikosti paketů vždy nějaké bajty zbudou nevyužity, připíší se do konta fronty a budou jí přičteny k nově přiděleným bajtům při dalším kole zpracování. Takto jsou fronty, které nejsou v jednom kole spravedlivě obslouženy (je z nich odesláno méně dat než by mělo být dle GPS), vykompenzovány v kole dalším.[3]

  Vrátíme-li se k příkladu z plánování WRR, můžeme stanovit kvanta dat

pro obsluhu ze stanovených vah jednotlivých front např. takto: $q_A = 100$, $q_B = 200$ a $q_C = 400$. Velikost paketů v každé frontě vezměme opět pro jednoduchost konstantní: $s_A = 64$ bajtů, $s_B = 1500$ bajtů a $s_C = 300$ bajtů. Počáteční stav deficitů jednotlivých front nastavme na nulu. V prvním kole obdrží fronta A na své obslužné konto 100 bajtů a je z ní tedy odeslán jeden paket o velikosti 64 bajtů a zbylých 36 bajtů jí zůstane na kontě jako deficit. Fronta B obdrží na konto dle své váhy 200 bajtů, což ale nestačí na odeslání ani jednoho paketu o velikosti 1500 bajtů, a proto není obsloužena a celých 200 bajtů je jí ponecháno na kontu pro další kolo. Fronta C obdrží 400 bajtů na konto a jelikož jsou v ní pakety o velikosti 300 bajtů, je jeden odeslán a 100 bajtů jí zbude na kontě. V druhém kole má fronta A na kontě 36 bajtů z minulého kola a dalších 100 je jí připsáno. Má tedy celkem pro toto kolo 136 bajtů, což už stačí na odeslání ne jednoho, ale dvou bajtů (a ještě jí 8 bajtů zbude). Tímto jí byla vykompenzováno neférové obsloužení z prvního kola. A takto systém DRR pokračuje dále.[3]

To, který plánovací algoritmus je implementován závisí na konkrétní aplikaci a jejích požadavcích. Z hlediska spravedlivosti dělení prospustnosti jsou preferovány algoritmy WFQ a DRR. Z hlediska složitosti implementace a nároků na výpočetní výkon, je naopak WFQ v nevýhodě oproti zbylým dvěma algoritmům. Hlediskem může být také omezení maximálního zpoždění, protože různé algoritmy dosáhnou různých výsledků, ovšem pro data citlivá na zpoždění bývají většinou implementovány prioritní fronty.[3]
Mimo vlastní plánování zpracovávání vstupních front si musí plánovač dávat pozor, aby nepřetížil výstupní rozhraní. To bývá většinou opatřeno FIFO zásobníkem, ze kterého jsou data teprve odesílána na výstupní rozhraní. Pokud plánovač dokáže obsloužit vstupní fronty rychleji než je linková rychlost výstupního rozhraní, pak by mohlo dojít k přeplnění výstupní fronty. Z tohoto důvodu je hlídáno zaplnění těchto výstupních front a při překročení určité hranice plánovač přestane odebírat data ze vstupních front, dokud neklesne zaplnění výstupní FIFO fronty pod určiou mez. Tento způsob řízení toku bývá nazýván "zpětný tlak". Jelikož tyto pravidla platí stejně pro všechny fronty včetně prioritní, musí s nimi být počítáno při návrhu systému a definici SLA.[3]

Zahazování (Dropping)

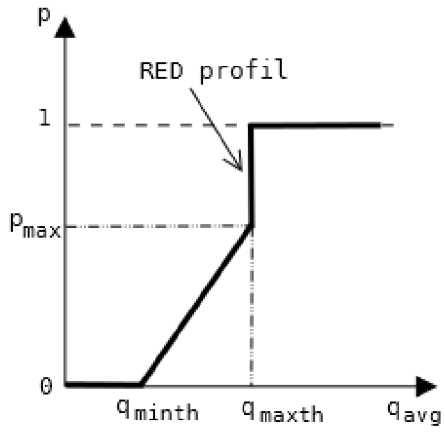
Jak už bylo zmíněno v přechozí sekci, směrovací prvky mají na vstupech většinou několik front, kde každá fronta pojme určité množství vstupujících paketů. Je-li rych-

lost příchodu paketů do některé fronty vyšší, než jakou je schopen plánovač danou frontu obsloužit, dojde za určitý čas nevyhnutelně k zaplnění této fronty (dosažení limitu maximálního zaplnění) a další příchozí pakety musejí být zahozeny. Kapacitu fronty by samozřejmě pro omezení zahazování paketů bylo možné navýšit, ovšem to by poté mělo negativní vliv na maximální zpoždění v této frontě. Maximální zpoždění ve frontě můžeme snadno spočítat, známe-li její kapacitu a rychlost obsluhy, podílem těchto dvou hodnot. Definujme si p_drop jako procentuální vyjádření pravděpodobnosti zahození paketu.[3]

- ***Tail Drop*** neboli zahození z konce fronty je jednoduše mechanizmus, který zajistí, že příchozí paket je zahozen, pokud se již do dané fronty nevejde. V případě této techniky je $p_drop = 0\%$ pokud je ve frontě pro paket dostatek místa. Pokud v ní dostatek místa není, je $p_drop = 100\%$.[3]
- ***Weighted Tail Drop*** tento mechanismus zavádí více různých limitů maximálního zaplnění, kde každý je přiřazen určitým paketům v závislosti na jejich označkování (například oranžové pakety z tříbarevného token bucket měření). Zahazováním vybraných paketů při dosažení nižšího limitu zaplnění tak lze předejít zahazováním jiných paketů, ke kterým se vztahuje vyšší limit zaplnění.[3]
- ***Random Early Detection*** (RED) je systém aktivního řízení fronty AQM (Active Queue Management), jehož úkolem je detekovat zahlcení sítě dříve, než dojde k zaplnění front. Takto je předcházeno nutnosti zahazovat pakety je zachována co největší efektivita sítě. AQM systémy bývají také označovány jako „congestion avoidance“, tedy techniky pro předcházení zahlcení. Algoritmus RED sleduje průměrné vytížení fronty a v případě nadměrného nárůstu o tom informuje koncové komunikující strany tím, že náhodně zahazuje některé pakety určitého datového toku. Omezí se tím i efekt tzv. globální synchronizace, který se projevuje u protokolu TCP díky jeho adaptivnímu chování. Pokud provoz TCP zahltí nějakou frontu, zjistí to tak, že jsou jeho pakety najednou ve velké míře zahazovány. Reaguje na to snížením rychlosti komunikace a tím i odstraní zahlcení front jeho pakety. Poté co zahlcení opadne, TCP opět zvyšuje rychlost přenosu, dokud opět nedostane zprávu o zahazování paketů. V případě uplatnění náhodného zahazování, které reaguje na blížící se zahlcení jemněji než technika tail drop, se zamezí tomuto cyklickému chování.[3]
  Průměrné vytížení fronty je vypočítáváno následnovně:

$$q_{avg} = q_{avg-old}\left(1 - \frac{1}{2^w_{red}}\right) + \left(q_{current}\frac{1}{2^w}\right) \text{[3]} \tag{1.3}$$

  kde $q_{avg-old}$ je průměrná hloubka fronty vypočítaná v předchozím kroku, $q_{current}$ je aktuální délka fronty a w_{red} je exponenciální váhová konstanta, jejíž hodnota určuje citlivost techniky RED na nárazové zatížení. Nižší hodnota w_{red}



Obr. 1.7: Pravděpodobnost zahození paketu technikou RED[3]

znamená, že se průměrné vytížení fronty více přiblíží aktuálnímu vytížení a zvýší se tedy pravděpodobnost zahazování paketů při nárazovém zatížení.[3]

- Pokud se q_{avg} nachází pod nastavenou minimální hranicí (q_{minth}), je paket zařazen do fronty.
- Pokud se q_{avg} nachází nad nastavenou maximální hranicí (q_{maxth}), je paket vždy zahozen. Toto zahození se nazývá "forced drop".
- Pokud se q_{avg} nachází mezi q_{minth} a q_{maxth}, bude paket zahozen s určitou pravděpodobností, kterou stanovíme takto:

$$p = \left(\frac{q_{avg} - q_{minth}}{q_{maxth} - q_{minth}} \right) P_{max} RAND(1)[3] \tag{1.4}$$

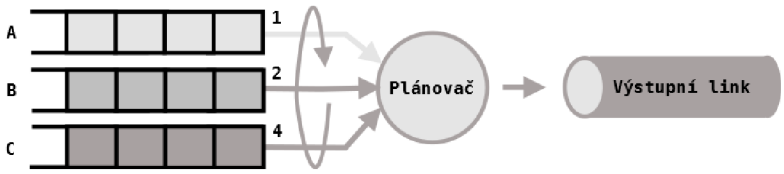
  kde p_{max} je pravděpodobnost zahození paketu přesně na hranici q_{maxth}, která ovlivňuje jak rychle roste pravděpodobnost zahození mezi hranicemi q_{minth} a q_{maxth}.

Závislost pravděpodobnosti zahození pakteu na nastavení různých hodnot algoritmu RED ukazuje obr.1.7[3]

- ***Weighted Random Early Detection*** (WRED) rozšiřuje systém RED o váhování analogicky k systému Weighted Tail Drop. Na jedné frontě je nastaveno více kategorií zahazování, kde má každá nastaveny jiné limity q_{minth} a q_{maxth} a operuje nad určitým druhem dopravy, který může opět rozlišovat dle různého značkování.[3]

Tvarování (Shaping)

Technika tvarování se velice podobá měření a omezování a také pro ni lze použít schéma tokenového kbelíku, které jsme si popsali dříve viz obr.1.4. Rozdíl nastává



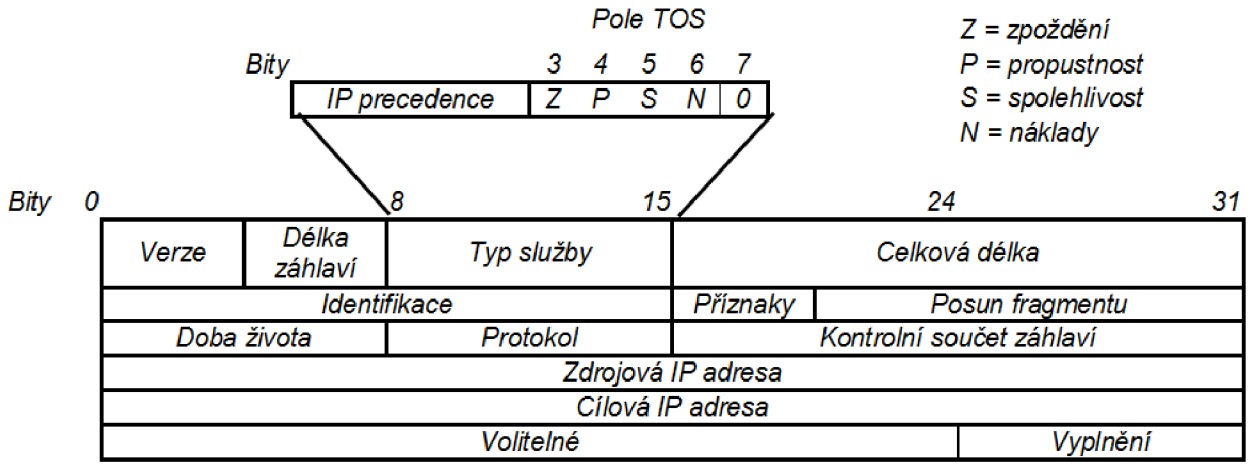
Obr. 1.8: Kombinace plánovače a tvarovače pro vícetřídní provoz[3]

v zacházení s pakety, pro které není v kbelíku dostatek tokenů. Zde nedochází k jejich značkování nebo zahazování, ale jejich zpracování je pouze odloženo. Paket tak čeká ve frontě a jakmile je v kbelíku dostatek bajtových tokenů odpovídající jeho velikosti, je zpracován. Technika tvarování takto vyhlazuje dopravní špičky a tvoří dopravu plynulejší, namísto aby při nárazovém přetížení pakety zahazovala.[3]

Pro tvarování bývá také využívána technika „leaky bucket“ neboli "děravý kbelík", kde kbelík neobsahuje tokeny, ale vlastní pakety, které z něj „vytékají“ otvorem ve dně stálou rychlostí. Příchozí pakety jsou shora plněny do kbelíku, jehož hloubka představuje maximální množství paketů (limit fronty). Pokud již není pro nově příchozí pakety v kbelíku místo, jsou zahazovány. Tento systém je využíván u ATM.[3] Tvarovač je v podstatě využíván k vyhlazování nárazové dopravy (dávek) a omezení propustnosti na určitou hodnotu. Může tak být využit pro agregaci datových toků více zákazníků na jednu linku nebo bývá také využíván pro řízení toku obsahujícího více kvalitativních tříd. Pro tuto aplikaci neobsahuje jen jednu frontu, ale více front dle počtu obsluhovaných tříd, které obsluhuje plánovač, jak je znázorněno na obr.1.8[3]

Linková fragmentace a prokládání

Při obsluze dat citlivých na zpoždění jsou tyto zařazovány do prioritní fronty a plánovač je zpracovává přednostně. Podle způsobu prioritního plánování je prioritní paket zpracováván např. po dokončení zpracovávání aktuálního neprioritního paketu. Pokud k tomuto přidáme výstupní FIFO frontu o velikosti 2 pakety, máme dohromady 3 pakety, které je potřeba odeslat na výstupní rozhraní před prioritním paketem. Pokud uvážíme velké 1500 bajtové pakety a nízkou rychlost linky, pak může zpoždění při odesílání v tomto jednom uzlu být nad přípustným limitem pro danou službu citlivou na zpoždění. Pro snížení tohoto zpoždění je při příchodu prioritního paketu možné neprioritní pakety ve výstupní frontě rozložit na fragmenty na linkové úrovni a ty následně při odelsílání postupně prokládat fragmenty prioritního paketu. V IP QoS se však tato technika příliš nevyužívá.[3]



Obr. 1.9: Umístění pole ToS v IPv4 paketu a jeho prvotní využití pro značky IP precedence a Type of Service[8]

1.4.3 Integrované služby (Integrated services)

V začátcích IPv4 bylo pro řízení kvality do paketů implementováno 8bitové pole Type of Service (ToS) - viz obr.1.9. Jeho první tři bity, pojmenované jako „IP precedence“ sloužily k označení důležitosti služby (8 hodnot) a další čtyři bity označovaly typ služby. Jednotlivé nastavení těchto bitů označovalo, zda-li je vyžadováno nízké zpoždění, vysoká propustnost, spolehlivost či nízká cena. Všechny tyto požadavky však měly pouze relativní charakter a jejich nastavení nezaručovalo dosažení konkrétních parametrů přenosu. Právě tyto nedostatky byly motivací pro vznik nového systému řízení kvality, který by umožňoval hladkou funkci službám vyžadujícím pro svůj chod určitou kvalitu přenosu.[3]

Technika integrovaných služeb, také označována jako „IntServ“ byla standardizována organizací IETF v dokumentu RFC1633. Pro zajištění kvality služby dle sjednaných SLA používá techniku rezervace potřebných zdrojů před vlastním zahájením datového přenosu služby pro každý datový tok. Protože tato rezervace musí proběhnout podél celé cesty sítí, musí být veškerá síťová zařízení schopná tuto rezervaci provést. To znamená, že musí znát potřebné mechanismy pro rezervaci prostředků a kontrolu přístupu (Admission Control). Kontrola přístupu hlídá jsou-li prostředky požadované pro nový přenos dostupné a na základě toho povoluje nebo zamítá rezervaci.[3]

Protokol RSVP

Pomocí tohoto protokolu jsou v architektuře IntServ vytvářeny jednosměrné rezervace síťových prostředků. Je-li vyžadován obousměrný přenos, pak jsou vytvořeny dvě samostatné rezervace. Rezervace probíhá následovně:

1. Vysílací strana odešle směrem k přijímací straně zprávu RSVP Path, která obsahuje:
   - množství prostředků vyžadovaných pro přenos datového toku
   - objekt ADPSEC, do kterého směrovače podél cesty k přijímači ukládají charakteristické informace QoS jako dostupné služby, odhady zpoždění, apod.
   - adresu předchozího skoku v síti, která je využívána pro nastavování zpáteční cesty od příjemce k odesílateli

2. Paket RSVP Path je směrován sítí pomocí standardního směrování. Každý router díky volitelnému poli záhlaví paketu Router Alert Option rozezná, že se jedná o zprávu protokolu RSVP. Pokud daný router nezná protokol RSVP, pak paket odešle standardním způsobem na další skok v síti dle své směrovací tabulky. Pokud jej zná, pak si pomocí obdržené adresy předchozího skoku v síti nastaví zpáteční cestu. Touto cestou pak bude směrován požadavek na rezervaci prostředků, jež následně odešle příjemcem směrem k odesílateli. Adresu předchozího skoku ve zprávě RSVP Path nastaví na svou adresu. Poté případně aktualizuje informace obsažené v objektu ADSPEC a zprávu odešle na další skok v síti standardním způsobem.[3]

3. Jakmile zpráva RSVP path dorazí k příjemci, jsou zanalyzovány informace z objektu ADSPEC. Na jejich základě příjemce případně upraví specifikaci rezervace, o kterou chce následně síť požádat. Pokud například směrovače v ADSPEC uveřejnili, že nemají dostatek zdrojů pro vysílání videa ve vysokém rozlišení, může přijímač snížit rezervační požadavek na video s nízkým rozlišením.[3]

4. Přijímač odešle směrem k vysílači zprávu RSVP Resv, obsahující:
   - požadovanou úroveň služby (např. služba s řízením zátěže)
   - objem přenosu specifikovaný dle schématu tokenového kbelíku
   - pětici informací identifikující datový tok (IP adresa a port odesílatele a příjemce a použitý protokol) sloužící ke komplexní klasifikaci [3]

5. Zpráva RSVP Resv je pak směrována sítí stejnou cestou, kterou byla směrována zpráva RSVP Path. Toto je zajištěno následováním zpáteční cesty, která byla v každém RSVP znalém směrovači nastavena zprávou RSVP Path. Každý RSVP směrovač zjistí, je-li příjemce oprávněn požadovat rezervaci prostředků. Pokud ano, pak zkontroluje, je-li možné rezervovat dostatek zdrojů pro datový tok dle požadavků v obdržené Resv zprávě, aniž by došlo k omezení stávajících toků. Je-li toto možné, pak jsou pro daný datový tok rezervovány zdroje, což může zahrnovat přiřazení prioritní nebo váhované fronty v plánovači dle požadované úrovně služby.[3]
   Pokud není možné požadavku na rezervaci vyhovět, router zašle příjemci zprávu ResvErr, čímž je rezervace zamítnuta. K tomu může dojít i pokud existuje alternativní cesta, ve které je dostatek prostředků pro přenos.[3]
   Prochází-li zpráva Resv přes nějaký router, který tento protokol nezná, je přeposlána jako obyčejný paket. S datovým tokem následně bude v těchto routrech zacházeno systémem best effort.[3]

6. Pokud zpráva Resv úspěšně dorazí až k odesílateli, ten se dozví, že je rezervace úspěšná a může začít s přenosem.[3]

Jelikož je RSVP protokolem typu "soft state", je nutné rezervaci periodicky obnovovat. Tak je zajištěna přizpůsobitelnost přenosu změnám v topologii sítě. Dojde-li k výpadku některé části sítě na rezervované cestě, je provoz přesměrován jinou cestou. Dočasně nejsou dodrženy požadované parametry služby, dokud není rezervace obnovena. Povinností rezervaci pravidelně obnovovat jsou zároveň automaticky ukončeny rezervace pro již neaktivní přenosy, které však nebyly řádně ukončeny. Řádným ukončením se rozumí zaslání zprávy PathTear nebo ResvTear odesílatelem nebo příjemcem.[3]
Architektura IntServ byla velkým pokrokem v řízení QoS, protože přinášela řešení typu end-to-end, tedy pro celou cestu sítí. Má však některé nevýhody, kvůli kterým nebyla příliš rozšířena:

- Do nespojově orientovaného internetu přináší spojově orientovaný model rezervace síťových prostředků.
- Prostředky jsou rezervovány, i když je zrovna daný datový tok plně nevyužívá.
- Poměrně vysoké nároky na režijní provoz, který se zvyšující se agregací datových toků neúnosně narůstá.

1.4.4 Diferencované služby (Differentiated services)

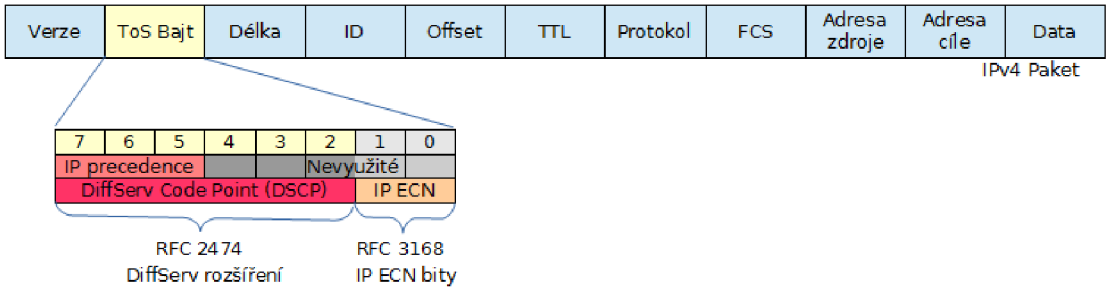
Tato architektura vznikla jako odpověď na výše uvedené nedostatky IntServ. Diferencované služby, nebo také zkráceně DiffServ, zajišťují QoS klasifikováním datových toků do různých tříd, ke kterým je pak přistupováno dle určitých pravidel. Část sítě, kde je DiffServ aplikováno, se nazývá doména DiffServ a její prvky musí tuto techniku znát. Klasifikace však probíhá pouze na hranicích DiffServ domény a směrovače uvnitř sítě se pak pouze řídí dle již nastavené třídy daného provozu. V současnosti se jedná o nejpoužívanější model v IP sítích.[3]
Funkce spočívá ve čtyřech základních krocích:

1. **Klasifikace** do jedné z definovaných tříd probíhá na vstupu do DiffServ domény pomocí implicitní, jednoduché nebo komplexní klasifikace. Tím, že počet tříd je omezen a chování ke každé z nich je předem domluvené a implementované do směrovačů, je značně omezeno zatížení sítě režijním provozem a procesováním ve směrovačích.
2. **Formování** pomocí technik omezování a tvarování je zajištěno, že přenos vstupující do DiffServ domény vyhovuje dohodnutým profilům - tzv. Traffic Conditioning Agreements (TCA), jež je odvozeno ze SLA.
3. **Značkování** probíhá pomocí pole DSCP (DiffServ Code Point) buďto na hranici DiffServ domény, nebo již při tvorbě záhlaví paketu. Informace v tomto poli pak slouží směrovačům uvnitř sítě ke zjištění, do které z tříd daný provoz patří.
4. **Řazení do front a plánování** probíhá na základě informací v poli DSCP tak, aby byly dodrženy SLA definované pro danou třídu provozu. Konkrétní způsob zacházení směrovačů s určitou třídou se nazývá Per Hop Behavior (PHB).[3]

Značka DSCP

Tato značka je umístěna v záhlaví paketu v 8bitovém poli zvaném Type of Service. Toto pole bylo v počáteční koncepci IP QoS původně určeno pro položky IP precedence a Type of Service, jak ukazuje obr.1.9. V řeči DiffServ je tento bajt nazýván DS pole.

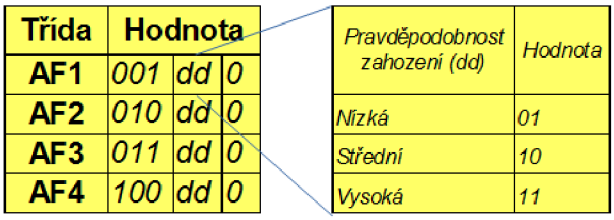
Strukturu DS pole ukazuje obr.1.10. Jejích prvních 6 bitů obsahuje vlastní DSCP značku, která slouží k rozlišení třídy provozu. Celkem 64 možných hodnot je rozděleno na 32 hodnot určených pro standardizované třídy dle IETF ($xxxxx0_2$) a zbylých 32 je pro experimentální účely. Níže uvedené uspořádání má charakteristiku doporučení a v případě potřeby může být poskytovatelem síťových služeb předefinováno. Každé definované třídě provozu je v síti přiřazen určitý PHB, který bude ve směrovačích pro všechny pakety dané třídy použit. Do jednoho PHB může být mapováno více DSCP hodnot.



Obr. 1.10: Umístění pole DS v IPv4 paketu na původní místo pole ToS a jeho využití pro DSCP se zachováním zpětné kompatibility k IP precedence [9]

PHB dělíme do těchto kategorií:

- **Standartní (default)** má hodnotu 000000_2 a je užit pro třídy, které nejsou mapovány do jiného PHB. K datovým tokům přistupuje systémem best-effort.
- **Urychlené předávání (Expedited Forwarding** – EF) má hodnotu 101110_2 a je určený pro aplikace vyžadující nízkou ztrátovost, zpoždění i jeho kolísání a zaručenou propustnost, jako např. hlasové služby. Pro takový PHB musí být zajištěno dostatek prostředků nezávisle na zatížení sítě jiným provozem, čehož je většinou dosaženo užitím prioritní fronty. Přetížení prioritní fronty je předcházeno technikou omezování. Jedině tak lze zaručit požadavky na propustnost a zpoždění. Aby byla zajištěna bezztrátovost, velikosti bufferů musí být větší než nárazový provoz charakteristický pro takto klasifikovanou dopravu.
- **Zajištěné předávání (Assured Forwarding** – AF) nachází využití pro aplikace vyžadující určitou minimální propustnost. Obsahuje sadu čtyř AF tříd (AF1 až AF4) a tří možných pravděpodobností zahození paketu (high / medium / low). Struktura DSCP pole pro tuto kategorii je na obr.1.11. Účelem takovéto struktury je poskytování služby v jedné ze čtyř definovaných tříd, kdy každá nabízí určitou přenosovou kapacitu. Pokud se zákazníkův objem dopravy drží v mezích nabízené třídy, jsou jeho data přepravována s nízkou pravděpodobností zahazování paketů. Toto je hlídáno pomocí techniky omezování. Pokud však objem dopravy překročí dohodnutou mez, zvýší se pravděpodobnost zahazování, což se uplatní např. při hrozícím přetížení sítě užitím technologie WRED.
- **Selektor tříd (Class Selector** – CS) má hodnotu $xxx000_2$. Jelikož tato kategorie využívá jen první tři bity pole DSCP, které se nacházejí na stejném místě v záhlaví paketu, kde byl dříve údaj IP precedence, použití těchto tříd je plně zpětně kompatibilní s původním systémem priorit IP precedence viz obr.1.10. Toto je také hlavním důvodem užití této kategorie PHB. Jednotlivým



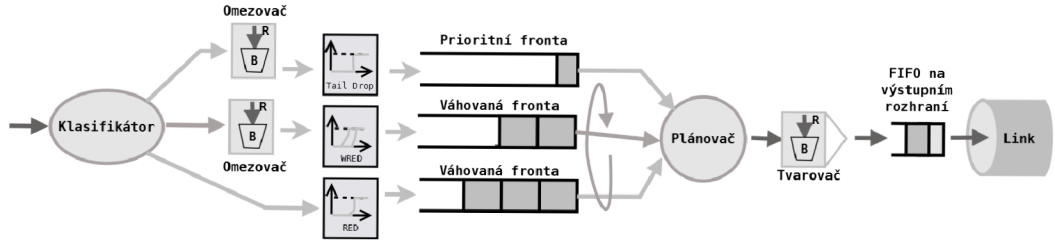
Obr. 1.11: Struktura DSCP pole pro kategorii AF[10]

priorităm jsou jednoduše přiřazeny určité parametry plánování a zahazování dle jejich důležitosti.

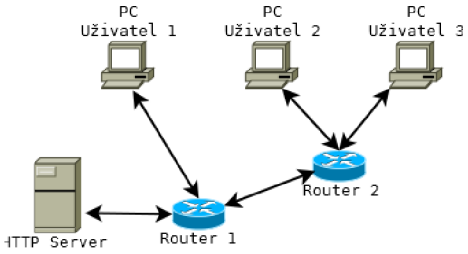
Poslední dva bity DSCP pole slouží pro přímé oznámení o blížícím se zahlcení sítě. Jsou nazývány Explicit Congestion Notification (ECN). Tato technika byla implementována do DiffServ za účelem zvýšit celkovou efektivitu sítě tím. Při blížícím se zahlcení sítě tuto skutečnost včas oznámí koncovým stanicím namísto zahazování jejich paketů (např. technikou RED). Tyto tak s předstihem mohou reagovat zpomalením rychlosti odesílání dat. Hodnotu ECN nastavují jednak koncové stanice, aby informovali síť o tom, zda-li hodlají (umějí) na přímé oznámení o zahlcení reagovat (00_2 = ne, 01_2 nebo 10_2 = ano) a jednak směrovače, aby případné zahlcení oznámili hodnotou 11_2.

1.4.5 **Implementace** QoS do směrovače

Výše zmíněné architektury určují systém, jakým jsou nástroje řízení kvality služeb sestaveny a vzájemně svázány, aby plnili svůj účel jako celek. Aplikace jednotlivých nástrojů do konkrétního směrovače záleží především na tom, ve které části síťové hierarchie se nacházíme. Typickou implementace na hranici přístupové sítě ukazuje obr.1.12 Je zde potřeba rozdělit síťový provoz každého zákazníka do tříd a agregovat datové toky více zákazníků na jednu výstupní linku, spojující přístupovou část sítě s distribuční částí.



Obr. 1.12: Typická implementace QoS do přístupového směrovače[3]



Obr. 2.1: Topologie simulované sítě

2 PRAKTICKÁ ČÁST

2.1 Návrh systému

Předmětem praktické části je tvorba softwaru umožňujícího simulovat a vyhodnocovat síťový provoz při různé konfiguraci systému QoS. Pro tyto účely je vytvořen model malé ethernetové sítě se třemi uživateli a jedním webovým serverem, kteří jsou vzájemně propojeni přes dva směrovače. Síť má stromovou topologii, kterou znázorňuje obr.2.1.

V této síti běží několik datových přenosů různých služeb. Pro přehlednost je vyhodnocován vždy jen jeden datový tok od každé služby a to ten nejzajímavější z hlediska sdílení přenosové kapacity s ostatními přenosy. Vše je přehledně zobrazeno ve schématu na obr.2.2.

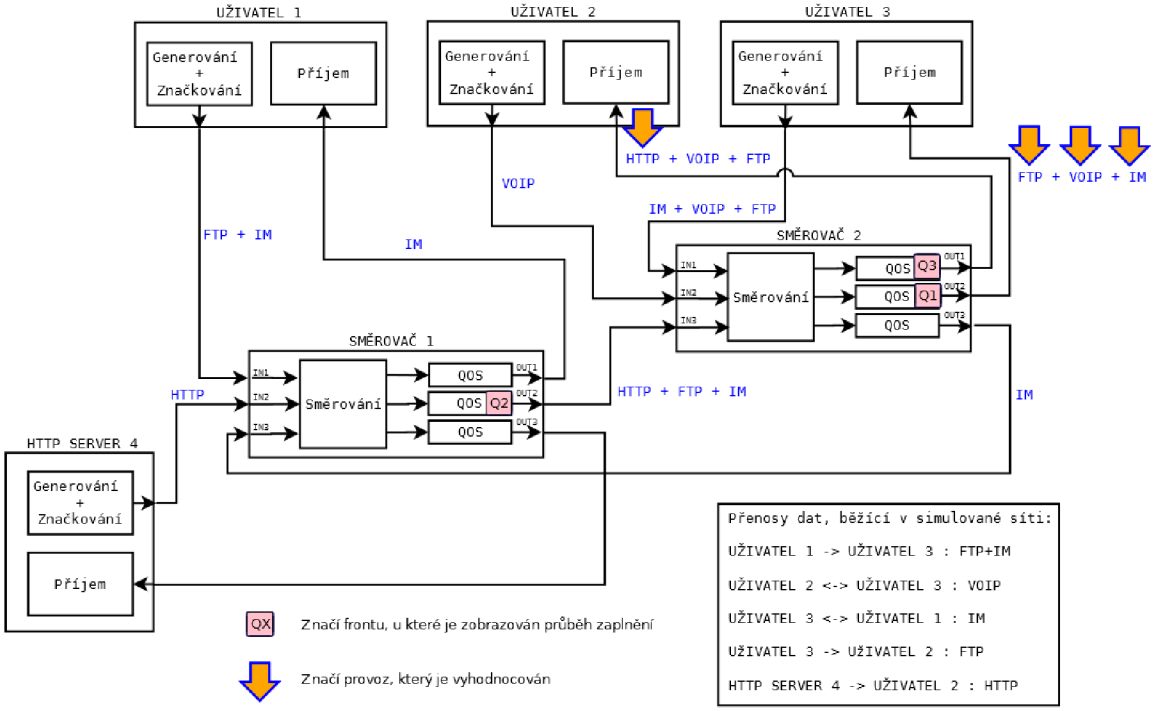
2.2 Realizace systému

Celý model je vytvořen v grafickém programovacím prostředí Simulink R2013a, které je nadstavbou Matlabu. Základní stavební bloky systému pocházejí především z toolboxu SimEvents, což je knihovna prvků určených pro diskrétní simulace. Pro přehlednost a také pro možnost opakovaného využití vytvořených funkčních celků v modelu jsou tyto zapouzdřeny do subsystémů. Náhled na blokovou strukturu modelu ukazuje obr.2.3.

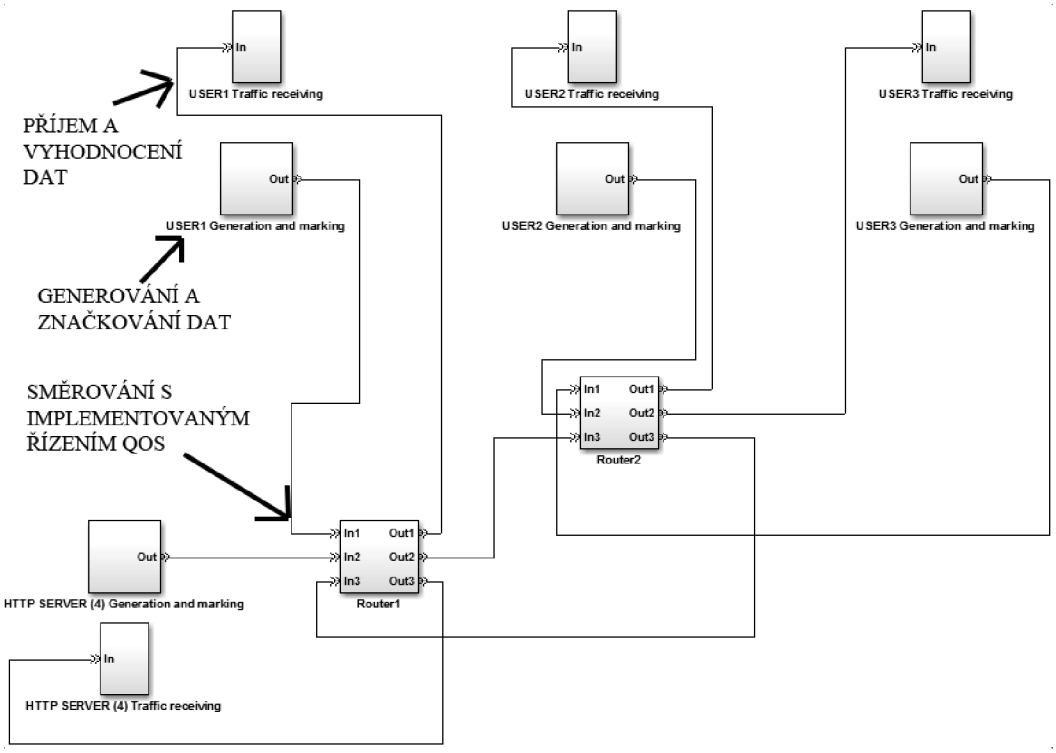
2.2.1 Subsystém pro generování a značkování paketů

Generování a značkování paketů zajišťuje pro každého uživatele subsystém "Generation and marking". Jeho vnitřní strukturu ukazuje obr.2.4.

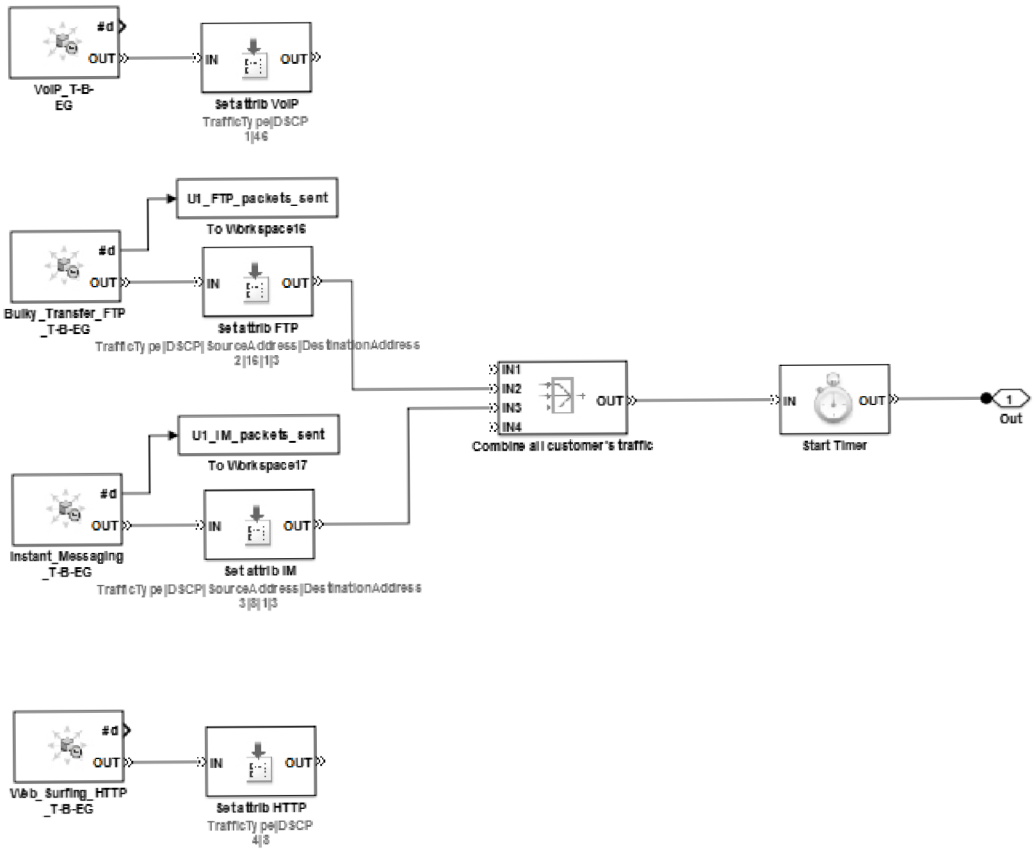
V každém tomto subsystému jsou nakonfigurovány generátory dat čtyř různých služeb, kdy rychlost jejich generování je u každé služby dána pevným rozsahem, ve



Obr. 2.2: Definovaný síťový provoz a vyhodnocované toky



Obr. 2.3: Bloková struktura modelu - nejvyšší úroveň



Obr. 2.4: Subsystém pro generování a značkování paketů

kterém se rychlost náhodně mění:

- VoIP – přenos hlasu v reálném čase - 400-1000 kbps
- FTP – přenos objemných souborů - 500-1000 kbps
- IM – online chat - 80-400 kbps
- HTTP – prohlížení internetových stránek - 200-2000 kbps

Je zde také zaznamenáván počet vygenerovaných paketů a každý paket je opatřen časovým razítkem pro pozdější vyhodnocení jeho zpoždění.

Značkováním je v datových jednotkách nastaveno pole DSCP, odpovídající jednomu ze tří PHB:

- Expedited Forwarding pro VoIP
- Class Selector 2 pro FTP
- Class Selector 1 pro IM a HTTP

K datovým jednotkám je také přidána zdrojová a cílová adresa, která bude dále využita ve směrovačích.

Jak je vidět na obr.2.4, tak nejsou některé generátory zapojené. Toto závisí na tom jaké provozy jsou v síti nadefinovány a liší se u jednotlivých uživatelů. U všech uživatelů je ale kompletní kopie tohoto subsystému se všemi generátory, aby bylo možné jednoduše přidat další datové toky.

2.2.2 Subsystém pro příjem a vyhodnocení dat

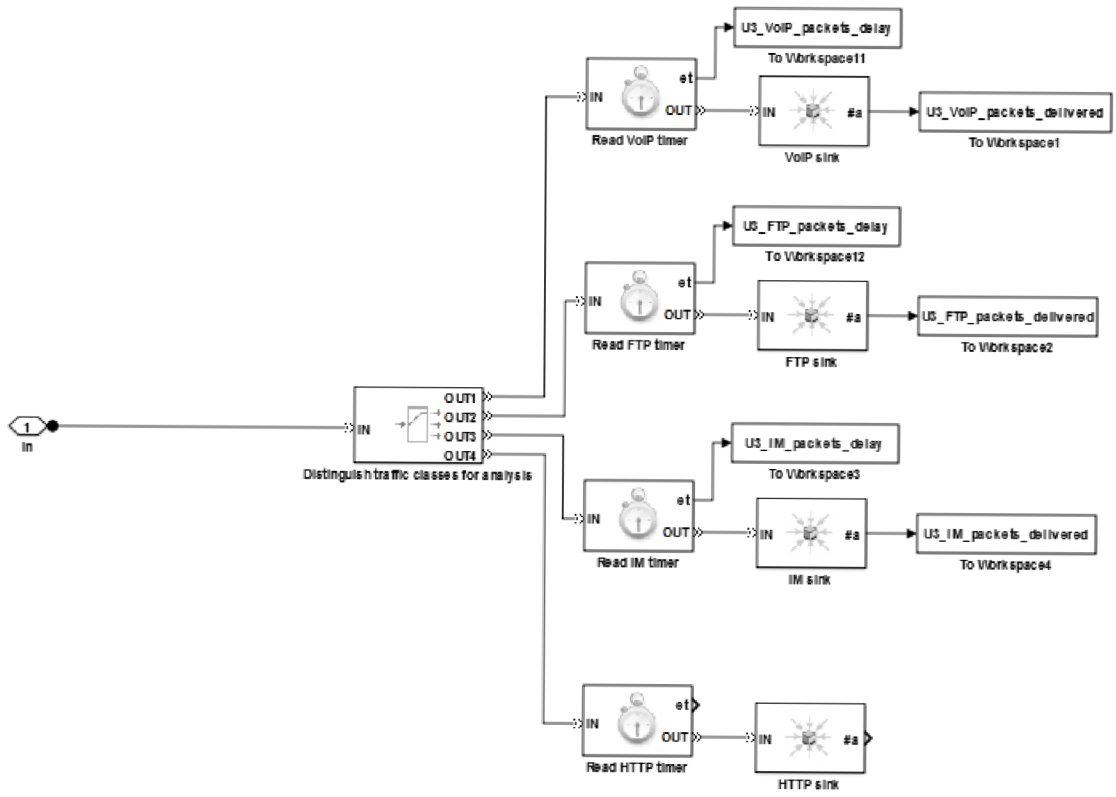
Příjem dat pro každého uživatele zajišťuje subsystém "Traffic receiving". Jeho vnitřní strukturu ukazuje obr.2.5.

Zde se zaznamenávají informace o počtu doručených paketů a jejich zpoždění. Záznam je prováděn jen pro aktuálně nadefinované datové toky, které jsou danému uživateli adresovány.

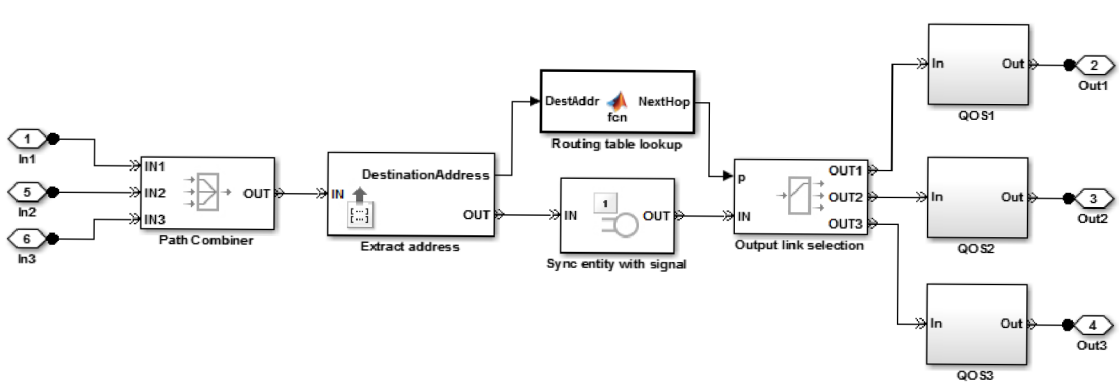
2.2.3 Subsystém směrovače

Blokové schéma směrovače je na obr.2.6.

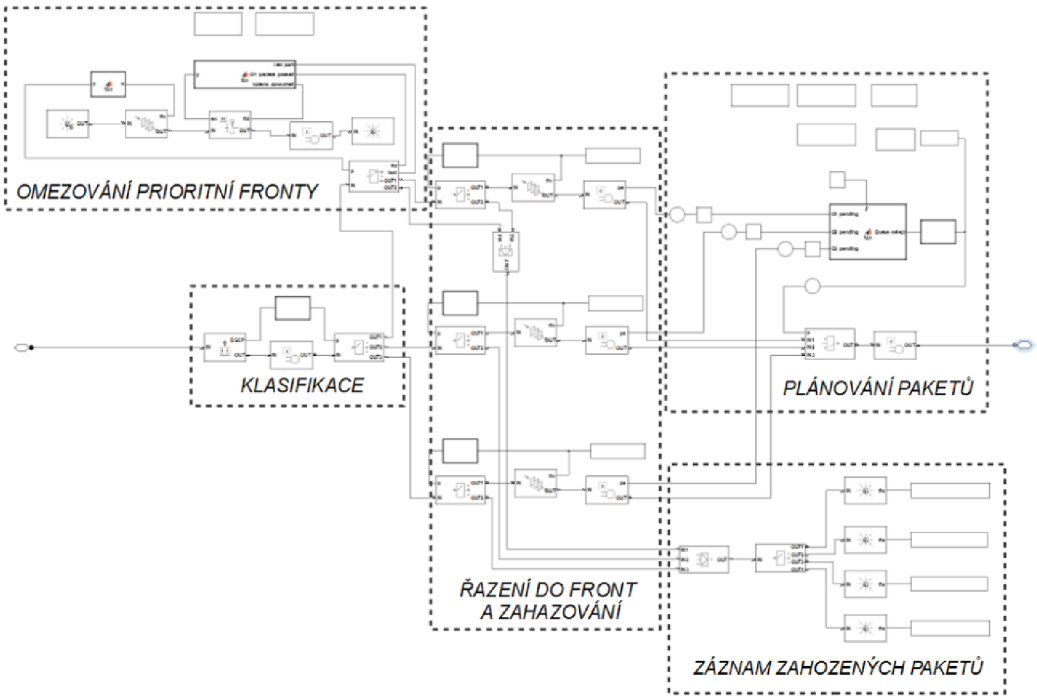
V první části směrovače je z každého paketu extrahována cílová adresa. Na jejím základě je pak ze směrovací tabulky vybrána adresa dalšího skoku v síti, respektive adresa výstupního rozhraní směrovače.
Na každém výstupním rozhraní je vřazen subsystém QoS, který bude podrobně rozebrán v následující sekci.



Obr. 2.5: Subsystém pro příjem a vyhodnocení dat



Obr. 2.6: Subsystém směrování



Obr. 2.7: Subsystém QoS

2.2.4 Subsystém QoS

Zde probíhá vlastní řízení datových toků dle stanovených pravidel. Na obr.2.7 je znázorněno členění celého subsystému na jednotlivé funkční celky.

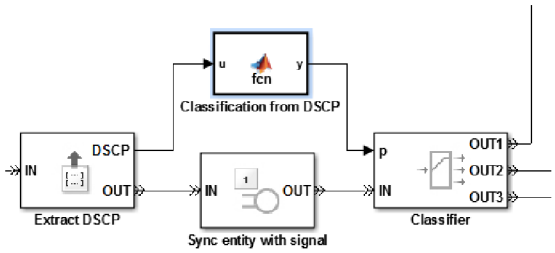
Klasifikace

Klasifikace probíhá na základě údaje v DSCP poli. Pro každý ze tří definovaných PHB je implementována jedna fronta:

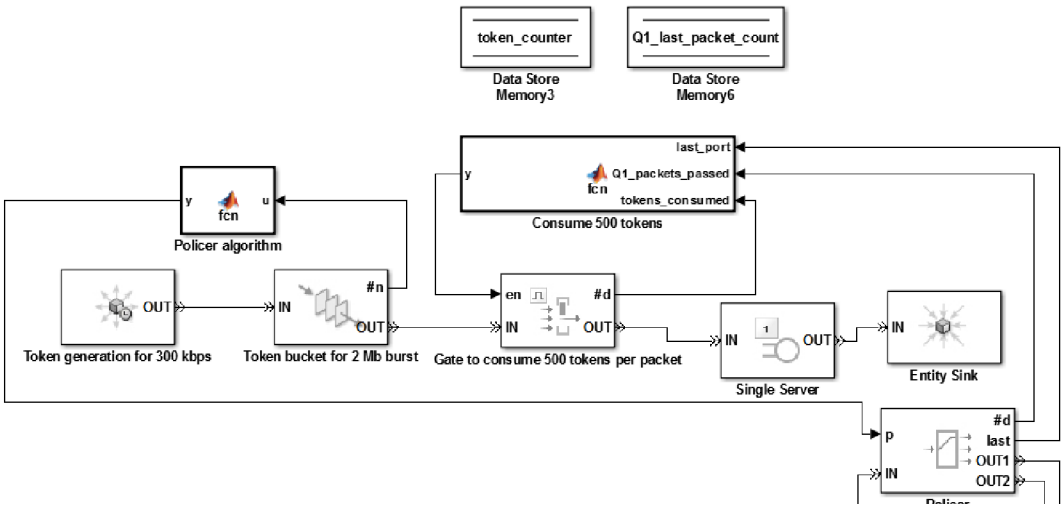
- Fronta Q1 pro třídu Expedited Forwarding, do které je mapován přenos VoIP
- Fronta Q2 pro třídu Class Selector2, do které je mapován FTP
- Fronta Q3 pro třídu Class Selector1, dle které je zacházeno s IM a HTTP přenosy

Omezování prioritní fronty

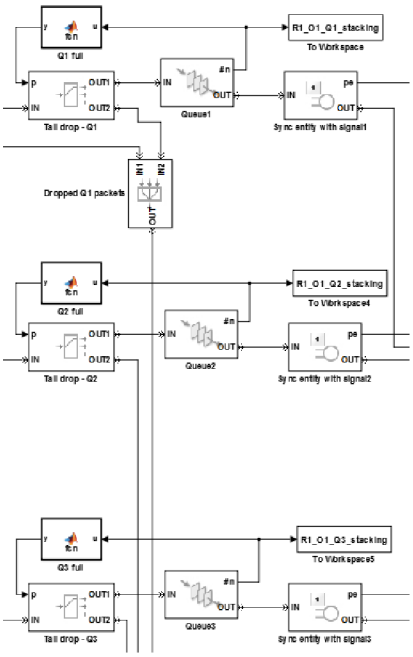
Před frontu Q1 je vřazen omezovač, který hlídá, aby tato fronta nezabrala při zvoleném prioritním plánování příliš velkou část přenosové kapacity výstupního rozhraní. Omezování je řešeno technikou tokenového kbelíku viz 1.4.2. Omezovač je přednastaven na propustnost 300 kbps a maximální dávku 2 Mbit. Implementace v modelu je vidět na obr.2.9.



Obr. 2.8: Klasifikace



Obr. 2.9: Omezovač prioritní fronty



Obr. 2.10: Systém front a zahazování paketů

Řazení do front a zahazování

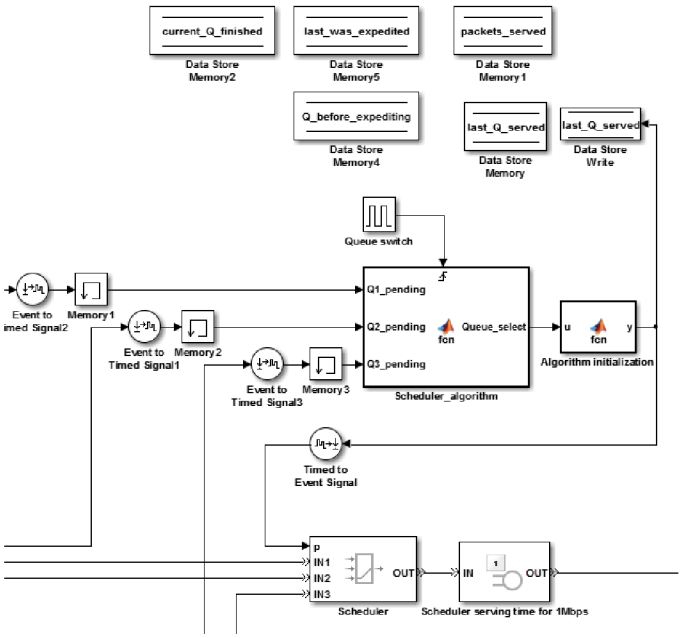
Model obsahuje tři FIFO fronty, kdy každá má buffer o kapacitě až 100 paketů. Skutečná kapacita fronty je nastavena před spuštěním simulace v rozmezí 1-100 paketů. Pokud dojde k zaplnění fronty, jsou další příchozí pakety zahazovány, dokud se ve frontě neuvolní místo. Je tedy použit systém zahazování tail drop. Blokové schéma této části ukazuje obr.2.10.

Plánování

Plánovač má za úkol přepínat data z jednotlivých front na výstupní rozhraní o přenosové kapacitě 1 Mbps. Jeho blokové schéma je na obr.2.11.

Samotný algoritmus, který určuje, ze které fronty budou data po odeslání aktuálního paketu zpracována, je implementován ve funkčním bloku Matlabu. Jeho chování se mění podle toho, které funkce QoS uživatel před spuštěním simulace nastaví:

- Šetření práce – Aktivací zajistíme vynechání prázdných front ze zpracování.
- Váhování front – Je-li zapnuté, je použit plánovací algoritmus WRR. Při vypnuté volbě jsou fronty obsluhovány cyklicky bez váhování.
- Prioritní plánování (EF) – Při zapnuté prioritizaci je fronta Q1 obsluhována prioritně, tedy vždy, když není prázdná, je ihned po dokončení zpracování aktuální fronty zařazena jako další ke zpracování. V základním nastavení je tedy prioritní zpracování nepreemptivní.



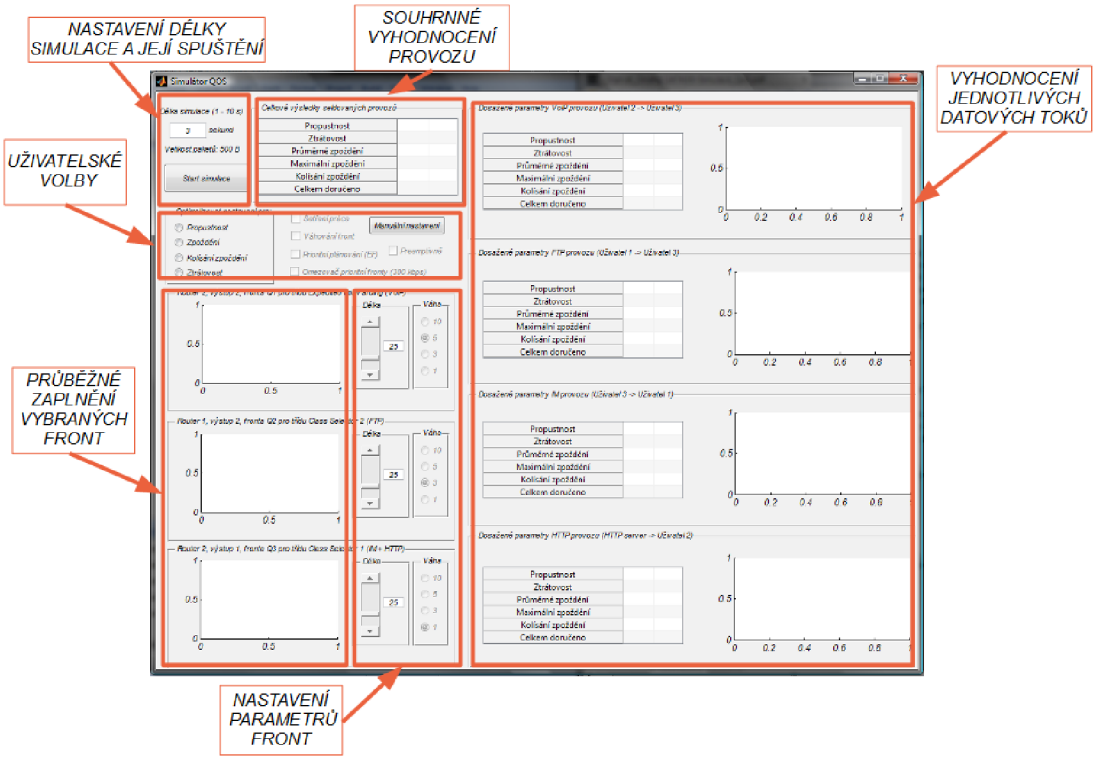
Obr. 2.11: Blokové schéma plánovače

- Preemptivně – Povyšuje prioritní zpracování fronty Q1 na preemptivní. To znamená, že pokud dorazí do fronty Q1 paket, tak je odeslán ihned po aktuálně odesílaném paketu a nečeká se na dokončení zpracování aktuální fronty. Ze své podstaty je tato volba relevantní jen pokud je zvoleno prioritní plánování a zároveň váhování. Bez váhování je totiž vždy zpracován jen jeden paket z dané fronty a tím je zpracování dané fronty ukončeno.
- Omezovač prioritní fronty – Omezí datový tok do prioritní fronty na průměrně 300 kbps. Největší povolená dávka je 2 Mbit.

2.3 Grafické uživatelské rozhraní

Uživatelské rozhraní slouží k ovládání celého programu a zobrazování výsledků simulace. Obsahuje jediné okno, které slouží jak k nastavování parametrů, tak i zobrazení výsledků simulace. Je tak vždy přehledně zobrazeno za jakého nastavení bylo zobrazených výsledků dosaženo. Popis jednotlivých částí uživatelského rozhraní je na obr.2.12.

Pro nastavení parametrů QoS systému jsou k dipozici dva módy. Po spuštění programu je aktivní automatický režim, ve kterém uživatel může vybrat který kvalitativní parametr je pro něj nejdůležitější, a software sám zvolí vhodné nastavení. Možnosti jsou:



Obr. 2.12: Grafické uživatelské rozhraní

- propustnost
- zpoždění
- kolísání zpoždění
- ztrátovost

Automaticky nastavené parametry jsou okamžitě po změně výběru zobrazeny na prvcích pro manuální nastavení. Po stisku tlačítka „Manuální nastavení“ se aktivuje plně manuální režim, kde uživatel může měnit veškerá nastavení včetně délky a váhy jednotlivých front. Uživatelské rozhraní vždy umožňuje nastavovat jen ty parametry, které mají v dané kombinaci s ostatním nastvením smysl. Pokud tedy například nezvolíme váhování front, není možné měnit váhy jednotlivých front.

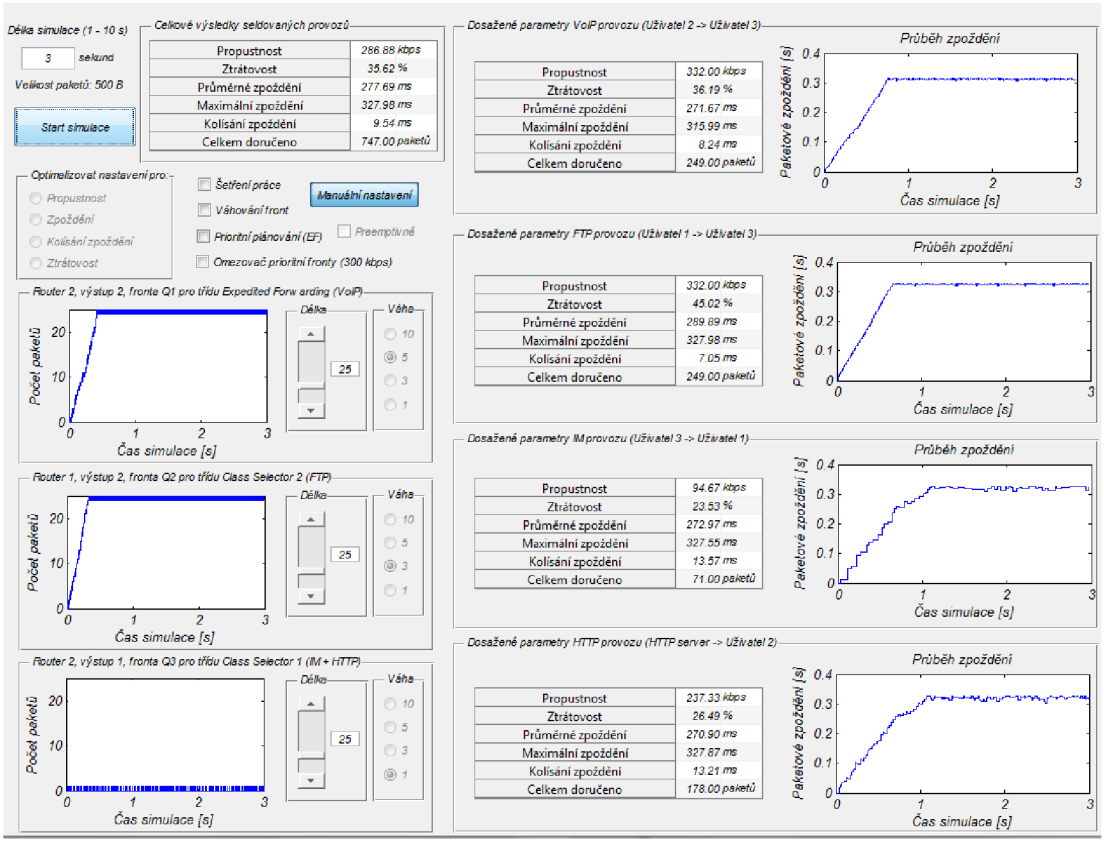
Pod sekcí uživatelských voleb je informativně zobrazeno průběžné využití vybraných front. Vybrány jsou fronty z takových rozhraní, která jsou nejvíce vytížena.

Nastavení front vedle těchto grafů však platí pro všechny fronty v celé síti a nevztahuje se jen na ty, jejichž zaplnění je zobrazeno.

Po ukončení nastavování můžeme ještě zvolit délku simulace. Doporučená délka je 1-10 vteřin s ohledem na výkon počítače, na kterém simulaci provádíme. Po stisku „Start simulace“ je provedena konfigurace modelu dle námi zvolených parametrů.

Poté proběhne vlastní simulace a po jejím skončení jsou vyhodnoceny dosažené výsledky a zobrazeny uživateli. Vyhodnocovány jsou tyto parametry:

- **Propustnost**
- **Ztrátovost**
- **Průměrné zpoždění**
- **Maximální zpoždění**
- **Kolísání zpoždění**
- **Počet doručených paketů**
- **Průběh aktuální hodnoty zpoždění během simulace**

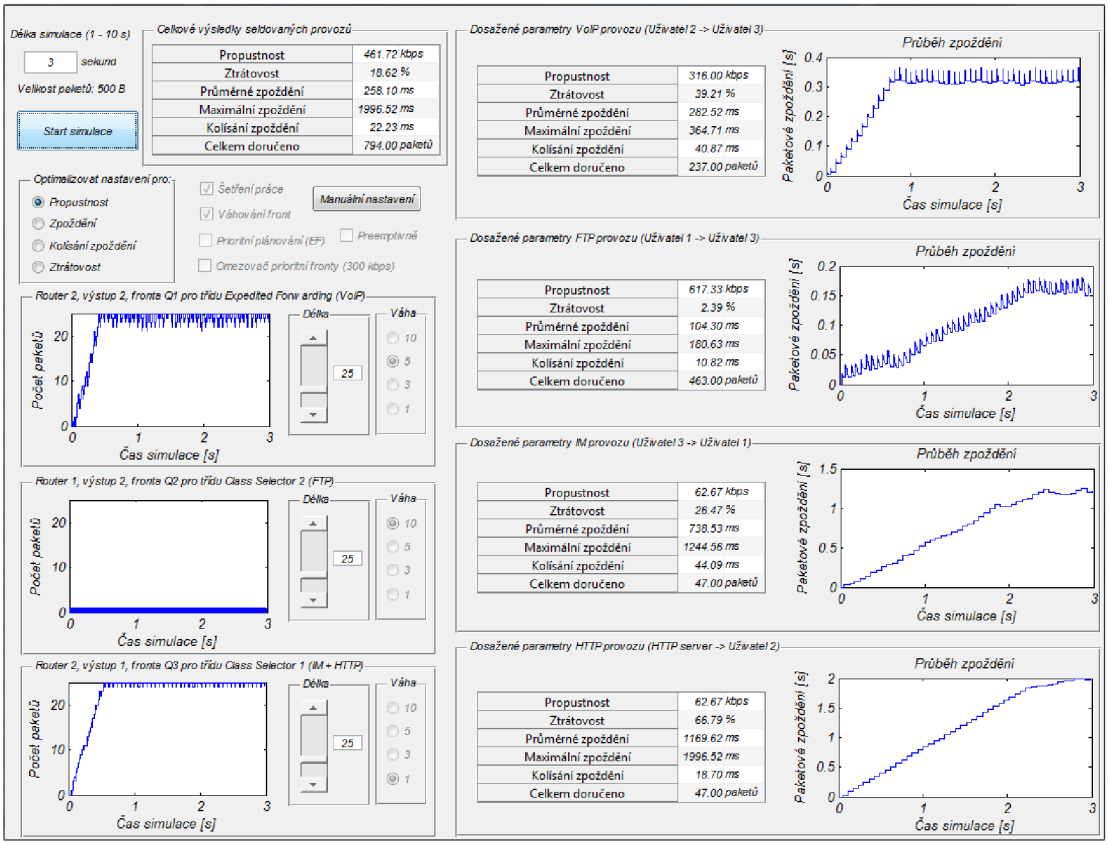


Obr. 3.1: Výsledky simulace při výchozím nastavení

3 VÝSLEDKY SEMESTRÁLNÍ PRÁCE

3.1 Simulace v základním nastavení

Výsledky simulace při základním nastavení, kdy je vypnuto váhování, prioritizace i šetření práce, ukazuje obr.3.1. Se všemi druhy provozů je zacházeno stejným způsobem. Jelikož jsou IM a HTTP provozy mapovány do stejného PHB, sdílí spolu jednu frontu, které plánovač věnuje jednu třetinu celkové přenosové kapacity výstupního linku. Druhá třetina kapacity je věnována VoIP a poslední FTP přenosu. Z výsledků je patrné, že toto nastavení není uspokojivé ani pro jednu z provozovaných služeb. Všechny přenosy se potýkají s vysokou ztrátovostí, která dosahuje až 45 % u FTP a zpožděním v řádu stovek ms. Tak vysoké zpoždění by mělo především pro VoIP nepříjemné důsledky, kdy by si komunikující strany díky zpoždění skákaly do řeči.

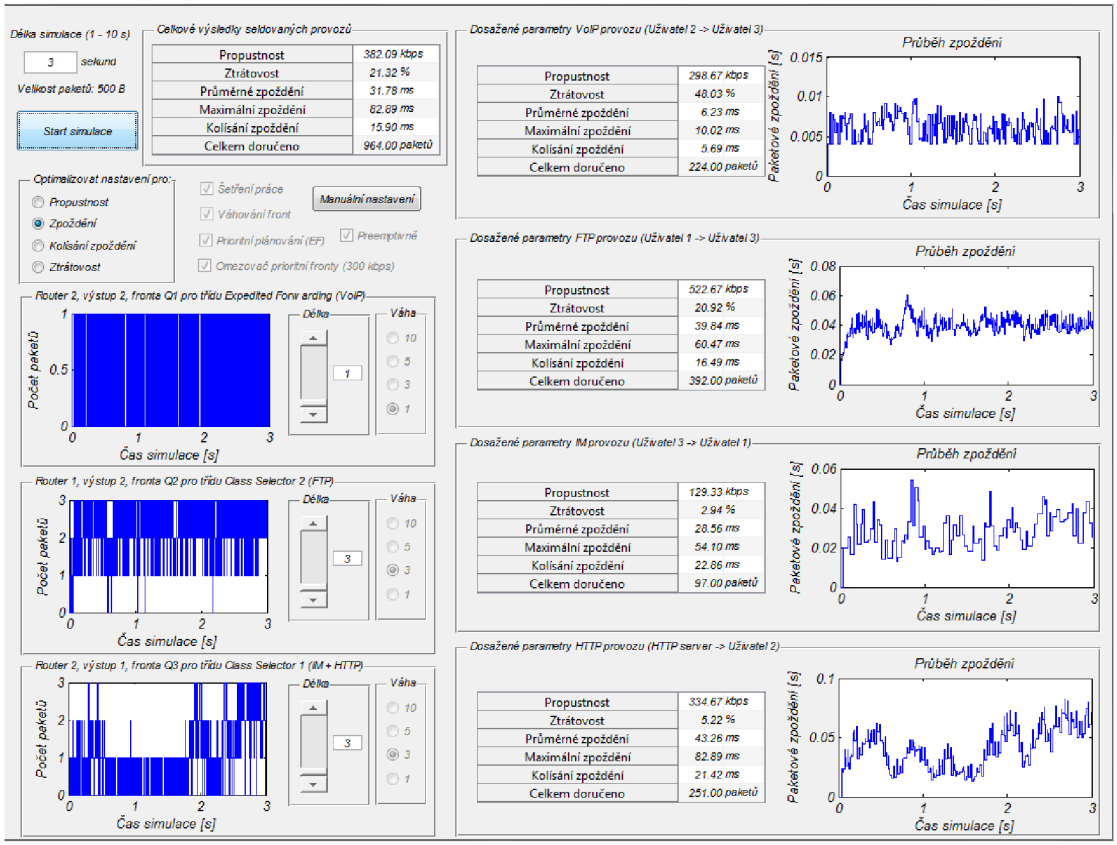


Obr. 3.2: Výsledky simulace po optimalizaci nastavení na vysokou propustnost

3.2 Optimalizace nastavení pro vysokou propustnost

Po zvolení optimalizace pro zvýšení propustnosti systém aktivuje volby šetření práce a váhování front. Váha front je s ohledem na rychlost generování dat jednotlivých služeb nastavena na 5 pro forntu Q1 (VoIP), 10 pro frontu Q2 (FTP) a 1 pro frontu Q3 (IM a HTTP).

Z výsledků simulace na obr.3.2 je patrné výrazné zvýšení celkové propustnosti z 286 kbps na 462 kbps. Tohoto zlepšení bylo dosaženo jednak díky šetření práce, protože se plánovač nezdržoval s prázdnými frontami, a jednak díky přiřazení vyšších vah frontám, které jsou více vytížené. Naopak u zpoždění jsme si příliš nepomohli. Hodnoty blížící se 300 ms jsou pro hlasové služby neakceptovatelné.



Obr. 3.3: Výsledky simulace po optimalizaci nastavení na nízké zpoždění

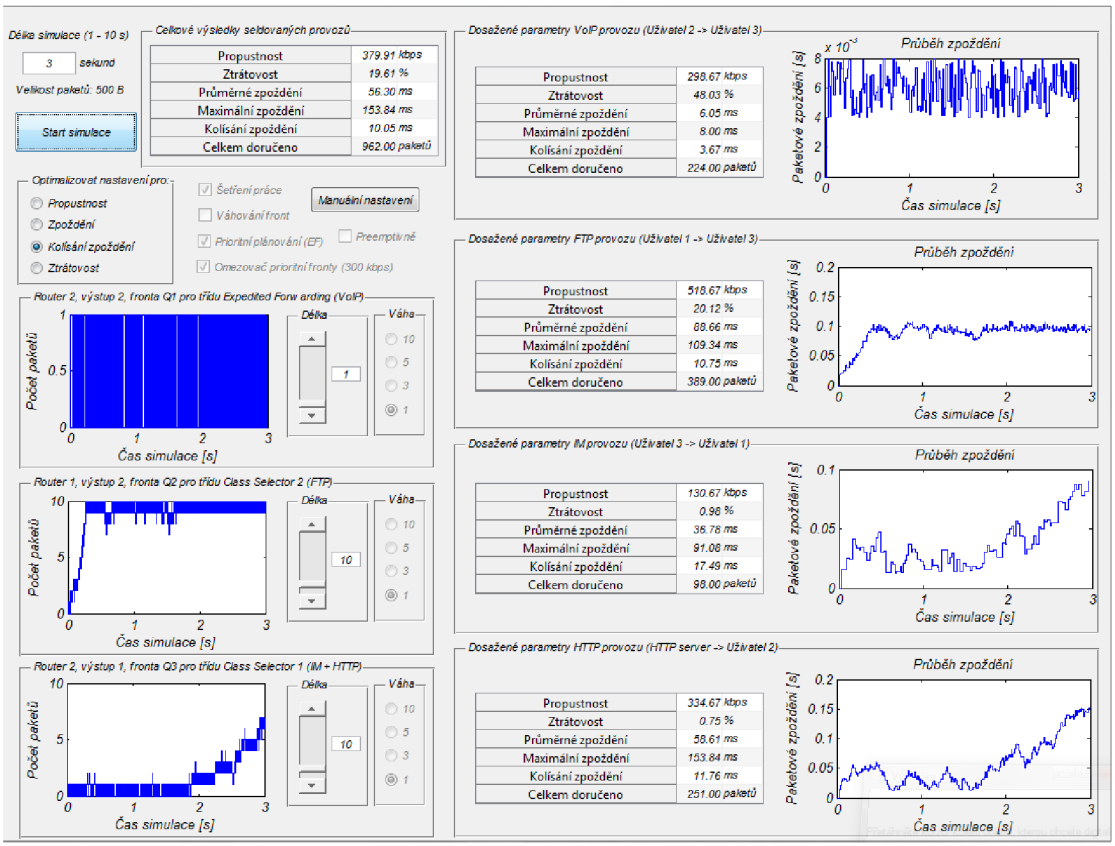
3.3 Optimalizace nastavení pro nízké zpoždění

Po změně preference na zpoždění je aktivována preemptivní prioritizace fronty Q1, aby především data citlivá na zpoždění byla doručena co nejrychleji. Kapacita všech front je snížena, protože pokud vyžadujeme nízké zpoždění, je lepší některé pakety zahodit, než je nechat čekat v dlouhé frontě. Je také zapnut omezovač prioritní fronty, aby při jejím velkém zatížení příliš nestrádaly ostatní provozy.

Efekt těchto změn je vidět jak u celkových výsledků, kde průměrné zpoždnění kleslo z 258 ms na 32 ms, tak především u výsledných parametrů VoIP provozu, kde zpoždění kleslo z 283 ms na 6 ms, viz. obr.3.3. Naopak ztrátovost a propustnost se zhoršily a to především díky častému zahazování paketů, které se nevejdou do krátkých front.

3.4 Optimalizace pro snížení kolísání zpoždění

Při použití sady nastavení pro nízké kolísání zpoždění je oproti předchozímu nastavení pro nízké zpoždění vypnuto váhování. Funkce váhování totiž způsobuje, že je z



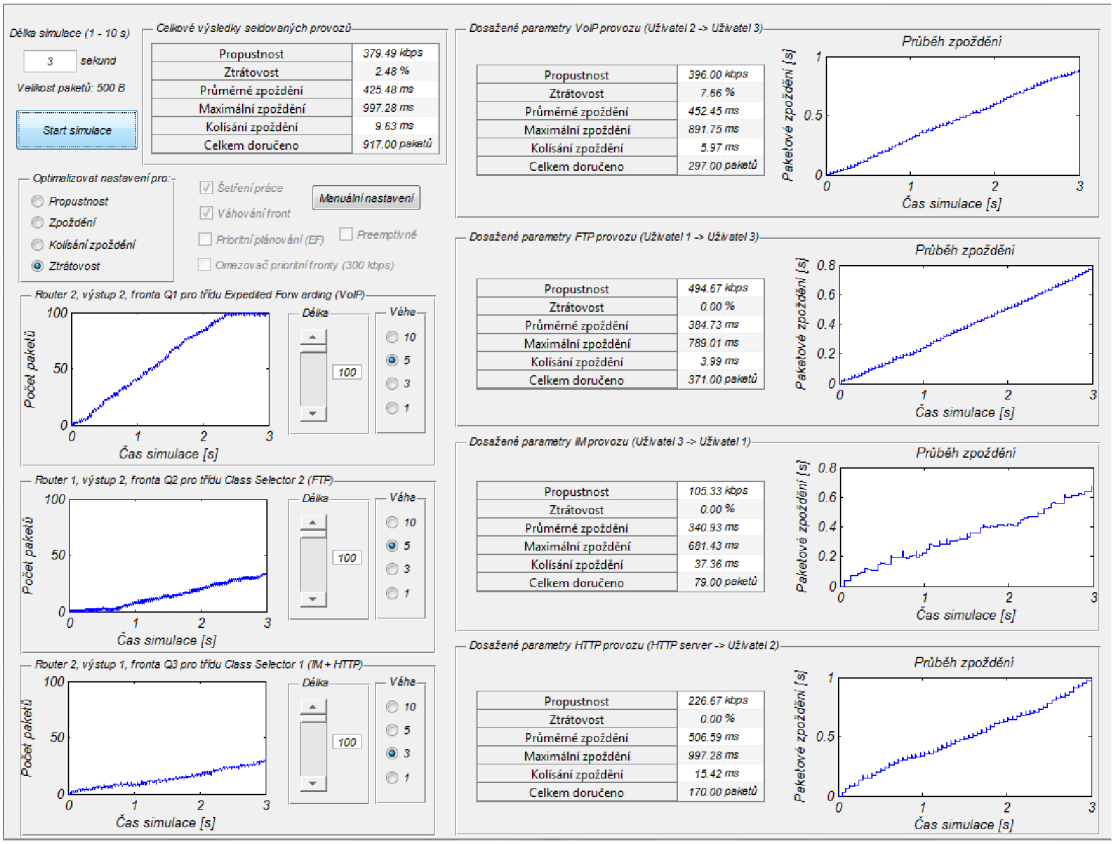
Obr. 3.4: Výsledky simulace po optimalizaci nastavení na nízké kolísání zpoždění

jedné fronty zpracováno více paketů za sebou, zatímco pakety v ostatních frontách čekají a roste jim zpoždění oproti předešlým paketům z této fronty. Když pak přijde řada na další frontu, je z ní opět zpracováno více paketů najednou. Ty jsou pak doručeny naopak s minimálním vzájemným zpožděním. Tento efekt by tedy způsobil velké rozdíly ve zpoždění po sobě následujících paketů. Dále je zvětšena kapacita front Q2 a Q3. Tím se vyhladí rozdíly zpoždění vznikající zejména, protože v nich řazené pakety procházejí oběma směrovači.

Už při nastavení pro nízkoé zpoždění bylo kolísání zpoždění hodně redukováno a to na 5,7 ms u VoIP a 15,9 ms celkově. Těmito úpravami parametrů bylo docíleno dalšího snížení kolísání zpoždění na 3,7 ms u VoIP a 10,1 ms celkově, jak ukazuje obr.3.4.

3.5 Optimalizace pro nízkou ztrátovost

Pro snížení ztrátovosti jsou vyřazeny nebo alespoň omezeny mechanismy pro zahazování paketů. Je tedy vypnut omezovač prioritní fronty a kapacity front jsou navýšeny



Obr. 3.5: Výsledky simulace po optimalizaci nastavení pro nízkou ztrátovost

na maximum. Váhování je aktivní a váhy jednotlivých front jsou nastaveny úměrně k jejich vytížení, aby se co nejvíce omezilo zahazování při zaplnění front.

Celková ztrátovost klesla při tomto nastavení na 2,5 %. Tento výsledek je však poměrně hodně zkreslen díky tomu, že simulace začíná s prázdnými frontami. Jelikož fronty mají nyní kapacitu nastavenou na 100 paketů, tak se nestihnou do konce simulace všechny naplnit a pakety, které v nich na konci simulace zbudou, nejsou zahozeny, ale ani doručeny.

4 ZÁVĚR

Bakalářská práce se zabývala tématikou zajištění kvality služeb v ethernetových sítích. Cílem bylo rozebrat tuto problematiku a na základě teoretických znalostí zkonstruovat systém pro provádění simulací zaměřených na optimalizaci QoS systému.

Během psaní teoretické části bakalářské práce jsem se hlouběji seznámil s parametry, které jsou v souvislosti s kvalitou přenášených služeb sledovány a jak je lze pozitivně ovlivňovat. Dále jsem rozebral techniky používané k řízení přenosů a tyto jsem následně aplikoval v praktické části do vytvářeného modelu.

Při tvorbě praktické části jsem se seznámil s prací v programovacím prostředí Matlab a Simulink a také se způsobem jakým lze obě tyto prostředí vzájemně propojit a předávat mezi nimi data. Vytvořil jsem model sítě, ve kterém jsou implementovány základní nástroje řízení QoS a probíhají v něm přenosy uživatelských služeb různých typů. Pomocí grafického uživatelského rozhraní lze měnit nastavení těchto nástrojů a sledovat vliv na výsledné parametry služeb.

Parametry je možné měnit buďto ručně jeden po druhém, nebo si zvolit, že chceme nastavení optimalizovat pro propustnost, zpoždění, kolísání zpoždění nebo ztrátovost. Ve druhém případě zvolí nejvhodnější nastavení software sám dle zvoleného kritéria a na výsledcích simulace vzápětí uvidíme efekt. Současně je také automatická konfigurace čitelná na ovládacích prvcích, takže uživatel vidí, jaké parametry software zvolil.

Simulace provozu začíná s prázdnými frontami ve směrovačích, což při nastavení velkých kapacit front poměrně značně zkresluje výsledky. Na druhou stranu je ale díky tomu možné sledovat závislost zpoždění paketů na postupně vzrůstajícím zaplnění front ve směrovačích. Pro tyto účely jsou graficky zobrazeny průběhy zpoždění u jednotlivých provozů a průběhy zaplnění nejvytíženějších front. Ačkoliv model obsahuje řadu zjednodušení, např. nepočítá se zpožděním při přenosu a nepočítá pakety, které po skončení simulace zbudou ve frontách, lze jej využít k demonstraci účelu a způsobu implementace základních nástrojů řízení kvality služeb.

LITERATURA

[1] NOVOTNÝ, V.: *Architektura sítí* Brno: Vysoké učení technické v Brně, 2011.

[2] ČÍKA, P.: *Multimediální služby* Brno: Vysoké učení technické v Brně, 2012. ISBN: 978-80-214-4443- 0. (cs).

[3] EVANS, J., CLARENCE, F.: *Deploying IP and MPLS QOS for Multiservice Networks* San Francisco: Elsevier, 2007. ISBN: 978-0-12-370549-5.

[4] VEGESNA, S.: *IP Quality of Service* Indianapolis: Cisco Press, 2001.

[5] IEEE COMPUTER SOCIETY: *IEEE Std 802.1Q, 2011 Edition - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks*, New York, 2011. ISBN 978-0-7381-6708-4

[6] CROWCROFT, J.:*Internetworking Multimedia* [online]. 1998, poslední aktualizace 03. 12. 1998. Dostupné z URL: <`http://www.cl.cam.ac.uk/~jac22/books/mm/book/book.html`>.

[7] The-Crankshaft Publishing:*Traffic Shaping and Policing* [online]. Dostupné z URL: <`http://what-when-how.com/ccnp-ont-exam-certification-guide/traffic-shaping-and-policing-congestion-avoidance-policing\discretionary{-}{}{}-shaping-and-link-efficiency-mechanisms/`>.

[8] Cisco press:*Optimizing Your Network Design* [online]. 2000. Dostupné z URL: <`http://www.cisco.com/cpress/cc/td/cpress/design/topdown/td0512.htm`>.

[9] Cisco press:*Enterprise QoS Solution Reference Network Design Guide* [online]. 2005. Dostupné z URL: <`http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/WAN_and_MAN/QoS_SRND/QoS-SRND-Book.pdf`>.

[10] Cisco press:*Introduction to IP QoS (Course* [online]. 1999. Dostupné z URL: <`http://docstore.mik.ua/cisco/pdf/routing/Cisco%20-%20Introduction%20to%20IP%20QoS%20%28Course%29.pdf`>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

QoS Quality of Services

SLA Service Level Agreement

TCP Transmission Control Protocol

UDP User Datagram Protocol

QoE Quality of Experience

MOS Mean Opinion Score

ITU International Telecomunication Union

ISO/OSI International Standards Organization/Open System Interconnection

TCP/IP Transmission Control Protocol/Internet Protocol

IEEE Institue of Electrical and Electronics Engineers

VLAN Virtual Local Area Network

TPID Tagged Protocol Identifier

TCI Tagged Control Information

VLAN-ID Virtual Local Area Network Identifier

CFI Canonical Format Indicator

DEI Drop Eligible Indicator

PCP Priority Code Point

MPLS Multiprotocol Label Switching

IETF Internet Engineering Task Force

RFC Request For Comments

RSVP Resource ReserVation Protocol

WRR Weighted Round Robin

WFQ Weighted Fair Queuing

C_X Kapacita fronty X

C_{out} Kapacita výstupní fronty

w_x Váha fronty X

s_{avgX} Průměrná velikost paketu

GPS Generalized Process Sharing

s_X Velikost paketu

r Počítadlo kol WFQ plánovače

$n_X i$ Pořadí zpracování paketu

DRR Deficit Round Robin

FIFO First In First Out

$p_d rop$ Pravděpodobnost zahození paketu

RED Random Early Detection

AQM Active Queue Management

q_{avg} Průměrná délka fronty

$q_{avg-old}$ Průměrná délka fronty z předchozího výpočtu

w_{red} Exponenciální váhová konstanta

q_{maxth} Maximální hranice fronty

q_{minth} Minimální hranice fronty

$q_{current}$ Aktuální délka fronty

WRED Weighted Random Early Detection

ATM Asynchronous Transfer Mode

IP Internet Protocol

ToS Type of Service

RAO Router Alert Option

TCA Traffic Conditioning Agreements

DSCP DiffServ Code Point

PHB Per Hop Behavior

DS Differentiated Services

ECN Explicit Congestion Notification

EF Expedited Forwarding

AF Assured Forwarding

CS Class Selector

IPDV IP Packet Delay Variation

VoIP Voice over IP

FTP File transfer Protocol

IM Internet Messaging

HTTP HyperText Transfer Protocol

SEZNAM PŘÍLOH

A SIMULAČNÍ SOFTWARE S GRAFICKÝM UŽIVATELSKÝM ROZHRANÍM

Hlavními soubory software jsou:

- diffserv_model.slx- model implementace QoS, vytvořený v prostředí Simulink R2013a
- GUI-main.fig - soubor grafického uživatelského rozhraní
- GUI-main.m - soubor se zdrojovým kódem funkcí, volaných z uživatelského rozhraní

Tento software je možné spouštět pouze na PC, kde je instalováno prostřední Matlab a Simulink R2013a včetně toolboxu SimEvents.