University of West Bohemia in Pilsen
Department of Computer Science and Engineering
Univerzitni 8
30614 Pilsen
Czech Republic

Dynamické směrování v překryvných sítích s využitím predikce

Odborná práce ke státní doktorské zkoušce

Jindřich Skupa

Technical Report No. DCSE/TR-2016-01
April, 2016

Distribution: public

Technical Report No. DCSE/TR-2016-01
April 2016

Dynamické směrování v překryvných sítích s využitím predikce

Jindřich Skupa

Abstract

This technical report consists of the current state of the art in overlay networks and dynamic routing with traffic prediction. Introduction to the overlay networks is followed by routing specifics in overlay networks. Subsequent chapters describe routing metrics and parameters measured in overlay networks. Following chapter introduces time series prediction algorithms and methods. The last chapter brings a description of the further work and the goals of the Ph.D. thesis followed by summary.

The work was supported by the UWB grant SGS-2013-029 Advanced Computer and Information Systems. (Pokročilé výpočetní a informační systémy).

Copies of this report are available on
http://www.kiv.zcu.cz/en/research/publications/technical-reports/
or by surface mail on request sent to the following address:

University of West Bohemia in Pilsen
Department of Computer Science and Engineering
Univerzitni 8
30614 Pilsen
Czech Republic

Obsah

1 Úvod

Obsahem této odborné práce ke státní doktorské zkoušce je popis aktuálního stavu v oblasti překryvných sítí (overaly networks) s důrazem na techniky dynamického směrování s různými metrikami, řízením kvality přenosu a predikce stavu překryvné sítě a jejích spojů. Na základě těchto poznatků budou definovány teze disertační práce a směr dalšího výzkumu.

Vzhledem k rostoucím nárokům aplikací na rychlost doručování obsahu, minimalizaci zpoždění a maximalizaci spolehlivosti, si překryvné sítě získaly pozornost odborné veřejnosti. Překryvné sítě jsou zkoumány s očekáváními optimalizujícími přenos tak, aby vyhověl nárokům aplikací a uživatelů na kvalitu síťových služeb. Klasické IP sítě nejsou schopny tyto požadavky řešit dostatečně, zvláště v globálním prostředí, kde panuje velká heterogenita a platnost různých nastavení, metrik nebo funkcí sítě končí s hranicemi autonomních systémů. Překryvné sítě tak vytváří prostředí pro aplikace, kde pro všechny účastníky platí stejná pravidla, pro která je překryvná síť budována.

V první části bude čtenář seznámen s technikou překryvných sítí a možnostmi využití, příkladem mohou být sítě pro doručování obsahu (Content Delivery Network, CDN), virtuální soukromé sítě (Virtual Private Network, VPN), anonymizační sítě (The Onion Routing, TOR), sítě pro zajištění spolehlivosti a kvality služeb (Quality of Service, QoS) a zajištění efektivity přenosu (skupinové vysílání, multicasting). Z výčtu možných použití je patrné, že překryvné sítě nalézají široké uplatnění v globální internetové síti a jejich využití je stále aktuální. Jejich hlavní výhodou je možnost použití různých technik (QoS, směrování) a možnost provozu samostatných služeb (doručování obsahu), které není možné v běžných sítích zajišťovat dostatečně efektivně nebo pro ně neexistuje přímá globální podpora.

Ve druhé části jsou popsány možnosti dynamického směrování v překryvných sítích podle jejich architektury a užití včetně využívaných algoritmů. Uvedeny jsou i návrhy možného dalšího využití dynamického směrování. Dynamické směrování v překryvných sítích probíhá na základě měřených metrik a jejich následném vyhodnocení a nalezení nejkratší vzdálenosti - minimalizace ceny přenosu. Tato část dále obsahuje i specifika dynamického směrování právě v překryvných sítích a jejich odlišnost od klasických sítí.

Následující část práce se věnuje predikci síťového provozu, zatížení linek a jejich dalších vlastností v čase. Vlastnosti síťového provozu jsou představeny jako časové řady, které je možné predikovat. Představeny jsou nejběžnější postupy pro predikci těchto řad a dále jsou prezentovány jejich vlastnosti a principy fungování.

Poslední částí je shrnutí mé činnosti v představené oblasti. Dále jsou představeny cíle budoucí disertační práce a základní předpoklady pro její tvorbu.

2 Překryvné sítě

Překryvné sítě pracují nad definovanou podkladovou vrstvou, v prostředí dnešních sítí je tou vrstvou protokol IP, případně některý z vyšších transportních protokolů TCP nebo UDP. Tento základní předpoklad nemusí platit pro všechny sítě, překryvné sítě lze vytvářet i nad protokoly nižších vrstev, příkladem je Multiprotocol Label Switching (MPLS)[46]. Překryvné sítě jsou navrhovány a budovány za cílem rozšířit aktuální funkčnost existujících sítí. Tím může být efektivní využití zdrojů podkladové sítě, zprostředkování přemostění jednotlivých sítí, které nejsou schopny mezi sebou běžně některou službu podporovat (multicasting). Dále také umožňují obcházet bezpečnostní a systémová nastavení daných sítí (bezpečnostní politika je natolik restriktivní, že omezuje uživatele v jejich přístupu k funkcím stávající sítě). Často jsou překryvné sítě budovány za účelem získat pro uživatele jistou výhodu, kterou by v klasické IP síti neměl, vzhledem k přístupu „nejlepší snahy“ doručit data, nebo porušení principu síťové neutrality. Princip fungování překryvných sítí je označován jako sobecký, jejich cílem je využít maximum dostupných prostředků podkladové sítě pro své uživatele a služby.

Existují různé přístupy k překryvným sítím. Podle jednoho je použití dané překryvné sítě pouze dočasné, do doby než bude technologie připravená a adaptovaná do podkladové sítě, například 6BONE zavádí IPv6 tunely přes existující sítě IPv4. Druhý považuje překryvné sítě za svébytnou technologii, která má dlouhodobé uplatnění a často její funkce ani není možné uspokojivě zakomponovat jako standard do vrstev podkladové sítě (P2P, TOR Onion routing, VPN, QoS).

2.1 Překryvné sítě v Internetu

Existence a realizace překryvných sítí v Internetu není nová a jeho prostředí funguje celá řada překryvných sítí různého typu a určení. Dnešní internet ostatně vznikl jako překryvná síť nad podkladovou telefonní sítí, následně se vyvinul do samostatně fungující infrastruktury a dnes jej telekomunikační služby začínají využívat jako podkladovou síť pro přenos hlasu a videa. Aktuálně většina překryvných sítí využívá Internet jako podkladovou vrstvu pro realizaci vlastních služeb. Struktura a velikost překryvných sítí také není nijak omezena a záleží na jejím návrhu a plánovaném využití. Struktura překryvných sítí často záleží na jejím primárním určení, typickým případem jsou P2P sítě uspořádané například do struktury stromu, kde je struktura vystavěna tak, aby bylo nalezení dat na co nejmenší počet skoků.

2.2 Rozdělení překryvných sítí

Překryvné sítě lze klasifikovat podle různých kritérií. Jedním z nich může být podle topologie (P2P, centralizované, hierarchické), další podle využití, určení nebo poskytovaných služeb (přechod od IPv4 k IPv6 - 6BONE, anonymizace TOR, multicasting MBONE, úložiště - CAN, DHT).

2.3 Podle topologie

Základem každé překryvné sítě jsou jednotlivé uzly podkladové sítě. Jednotlivé kroky cesty mezi uzly A a B jsou pro překryvnou sítí považovány za jeden spoj. Obecně mohou uzly komunikovat každý s každým - tvoří úplný orientovaný graf, pro další použití se pak dle účelu organizuje překryvná síť do konkrétní topologie. Tato transformace nemusí probíhat vždy a pokud probíhá, pak může mít charakter trvalý (pevné uspořádání) nebo dočasný (dynamické uspořádání podle aktuálních potřeb). Překryvné sítě se nejčastěji přeuspořádávají za účelem poskytnutí lepších služeb a vyvažování zátěže - například škálování poskytovaných zdrojů, distribuce a redistribuce poskytovaných služeb nebo vyhledávání a replikace obsahu. Ke změně struktury sítě může dojít také vlivem vnější změny, například změna parametrů propojovacích linek nebo parametrů jednotlivých uzlů nebo také přidáním nového uzlu. Některé překryvné sítě mohou mít strukturu pevnou a po dobu jejího fungování neměnnou.

Vzhledem k tomu, že topologie překryvné sítě není nijak závislá na fyzické topologii nebo jejích možnostech, může se jednat o běžné topologie z klasických sítí, peer-to-peer, hvězda (centrální uzel, přes který ostatní komunikují, nebo má centrální databázi uzlů, zdrojů atp.), strom (stromová struktura jednotlivých komunikačních spojů, např. multicasting), kruh, kombinace předchozích, matice nebo n-rozměrná kostka (DHT, CAN). Jaké topologie se používá pro jaké určení, bude popsáno v následujících kapitolách popisujících existující sítě.

2.4 Podle využití

Jak bylo již uvedeno v předchozích sekcích, využití překryvných sítí je široké. Vytvořením překryvné sítě je možné sledovat nebo poskytovat mnoho síťových služeb. Od klasických parametrů přenosu jako je doručení dat, rychlost komunikace, spolehlivost, zaručení kvality služeb, přes služby koncových uzlů poskytující úložný prostor nebo výpočetní výkon až po služby sloužící k anonymizaci uživatelů a šifrování komunikace. Překryvná síť může sledovat více těchto cílů zároveň nebo některé jsou prostředkem pro dosažení jiného, například:

- komunikační infrastruktura,

- spolehlivost doručení dat,
- kvalita spojení (latence, propustnost, stabilita),
- úložiště nebo poskytování dat,
- poskytování výpočetního výkonu,
- poskytování aplikačních služeb,
- bezpečnost,
- anonymizace.

2.4.1 Komunikační infrastruktura - směrování

Překryvná síť, jejímž určením je nabízet komunikační infrastrukturu, může sledovat několik různých metrik. Například je navržena za účelem zvýšení spolehlivosti doručení dat, to provádí tak, že detekuje a propaguje výpadek rychleji než klasické IP protokoly. Bezodkladně zajišťuje alternativní záložní cestu. Internetový směrovací protokol - Border Gateway Protocol (BGP) pro směrování mezi autonomními systémy (AS) má vzhledem k rozlehlosti dnešního internetu relativně dlouhé konvergenční časy, než je výpadek detekován a náhradní cesty jsou přepočítány v celé síti. Cílem návrhu BGP byla hlavně stabilita a schopnost směrovat rozlehlé sítě.

Překryvné sítě pracují obyčejně s menšími směrovacími tabulkami, mohou předpočítávat alternativní cesty a rychleji detekují výpadky. Doba výpadku je tímto postupem minimalizována a koncové uzly jej nemusí být schopny ani detekovat. Překryvná síť zajistí transparentní přesměrování provozu. Kromě rychlého přepínání síťových cest může její konstrukce být navržena tak, že provádí korekci podkladových protokolů, kde nastavení metrik a použitých cest nemusí být z hlediska síťového provozu optimální, ale mohou být ovlivněny ekonomickými nebo politickými faktory.

Směrování provozu v internetu je výsledkem dvoustranných dohod jednotlivých síťových operátorů nebo politikou předávacích (peeringových) center. Některé linky tak mohou být znevýhodněné například kvůli jejich provozní ceně. Některé linky mohou být také využívány pro různé specifické účely a jejich parametry mohou být využity i jinak. Řada linek je také privátních, přenáší například data mezi pobočkami a nepoužívá se pro směrování veřejného datového provozu. Překryvná síť je schopna tento stav efektivně překrýt takovými cestami, které jsou skutečně pro daný typ provozu a její konfiguraci výhodné. Překryvné sítě mohou také reagovat na aktuální stav linek, jejich využití a dostupnou propustnost. Často zajišťují řízení kvality přenosu a rezervaci zdrojů po trase, hojně jsou pak využívány při přenosu proudu dat audio-video (VoIP, IPTV).

Překryvné sítě mohou také umožnit mobilitu zařízení. Adresování uzlů v Internetu je realizováno IP adresami, které jsou přidělované v rámci sítí. Aby byl klient pod danou IP adresou dostupný, musí v dané síti fyzicky existovat, nebo tam mít

zástupce. Překryvné sítě umožní spojení i s uzlem, který je aktuálně v jiné síti. V klasické IP síti to lze realizovat buď kombinací překladu adres (NAT) v kombinaci například s virtuální privátní sítí - VPN (dle potřeby). Varianta s NAT bez VPN lze použít v případech, kdy známe nové umístění klienta, při použití VPN se statickou adresou lze tuto konkrétní adresu překládat nebo směrovat do VPN bez ohledu na to, kde mobilní uzel zrovna je. Dalším řešením je právě využití překryvné sítě, kdy IP adresa je službou překryvné sítě a je skrze ní směrována k mobilnímu uzlu. Oproti použití klasické VPN lze v překryvné síti využít jejích služeb využívající optimalizace datových tras jak bylo uvedeno v první části sekce.

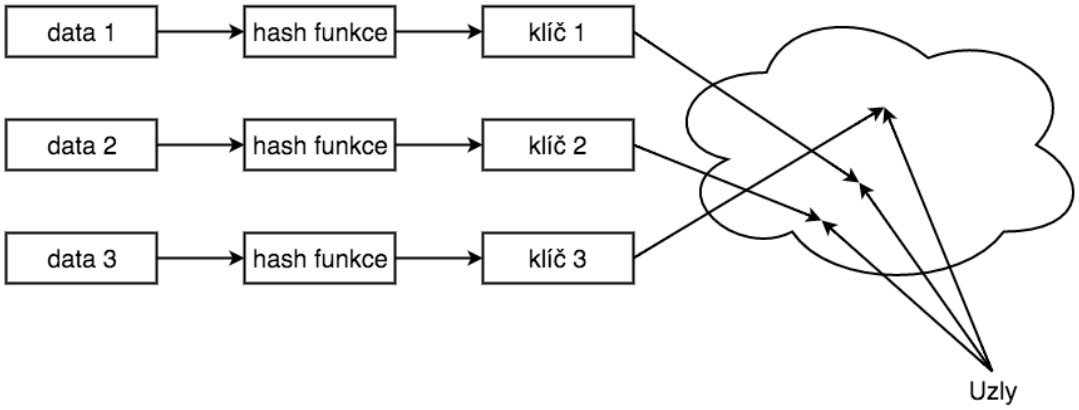
2.4.2 Ukládání a sdílení dat

Překryvné sítě navržené pro ukládání dat lze rozdělit podle způsobu užití na klasické P2P sítě určené pro sdílení velkých souborů a na content delivery networks, které ukládají obyčejně velké množství menších souborů. Klasické P2P sítě pro sdílení souborů také nekladou vysoké nároky na rychlost přenosu. Proti tomu CDN jsou zřizovány právě za účelem distribuovat data k uživateli na co nejkratší vzdálenosti a co nejrychleji (geografická dostupnost dat, poplatky za tranzitní přenosy, zpoždění a rychlost přenosu). Dalším rozdílem je, že v případě P2P sítí pro sdílení souborů je uživatel této sítě její součástí, je členem sítě a také poskytuje svoje zdroje ostatním. V případě CDN přistupuje uživatel ke zdrojům sítě zprostředkovaně, přes hraniční uzly CDN sítě.

Klasické P2P sítě, označované jako první generace, např. Napster, většinou obsahovaly jeden centrální uzel, který udržoval informace o všech sdílených datech a jejich umístění. Tento přístup se ovšem ukázal jako nedokonalý a naivní, protože centrální uzel, byť replikovaný, se pro síť stává kritickým a v případě jeho výpadku je ochromen provoz celé sítě. Proto byly navržené P2P sítě druhé generace, které eliminují tento centrální prvek a vše je plně distribuované, příkladem těchto sítí jsou sítě využívající distribuovanou hashovací tabulku (DHT), jejichž implementací existuje celá řada, CAN (Content Addressable Network), Chord, Kademlia, Pastry, Riak, Tapestry. Některým bude věnován prostor v následujících kapitolách.

Základní princip fungování DHT je popsán na obrázku 1. Bloku dat je na základě hashovací funkce přiřazen klíč, který zároveň určuje, na kterých uzlech budou data uložena. Uzly mají na základě svého unikátního identifikátoru přiřazeno umístění v prostoru klíčů stejně jako data. Klíče jak uzlů, tak dat jsou ze stejného prostoru a udávají jejich polohu. Prostor je podle konkrétní sítě reprezentován v 2D prostoru nebo v n-dimenzionálních útvarech (kostkách). Prostor klíčů lze reprezentovat i stromem. Ukládaná data jsou přenesena na některý ze známých uzlů (sousedů), který je nejblíže ke klíči dat (dle metriky daného prostoru). Ten přenese data na dalšího souseda blíže ke klíči. V konečném čase jsou data uložena

na uzlu, který je globálně nejblíže klíči dat. Přístup k datům se pak děje stejným způsobem. Pokud chce uzel získat daná data, kontaktuje svého souseda nejblíže klíči, ten vyhledá dalšího svého souseda blíže klíči nebo poskytne data ze svého úložiště. Tento způsob směrování bývá označován jako směrování klíčem (key-based routing). Seznam sousedů je obyčejně jednoduchý slovník obsahující pozici známých uzlů a jejich IP adresy, nesouvisí nijak s blízkostí uzlů na síťové úrovni. Přenosové cesty trasované k datům tímto způsobem ovšem nemusí být optimální z pohledu přenosové rychlosti. Seznam sousedů je možné udržovat systematicky tak, aby sousedé byly vybíráni podle vzájemné přenosové rychlosti (přenos dat) nebo za účelem rovnoměrného pokrytí prostoru klíčů (rychlost vyhledávání). Po nalezení uzlu, který má data právě k dispozici, lze použít směrování pro přenos dat z předchozí kapitoly, kdy sledujeme dosažení maximální přenosové rychlosti. Distribuce umístění a seznamu sousedů může být objektem vědeckého zkoumání.



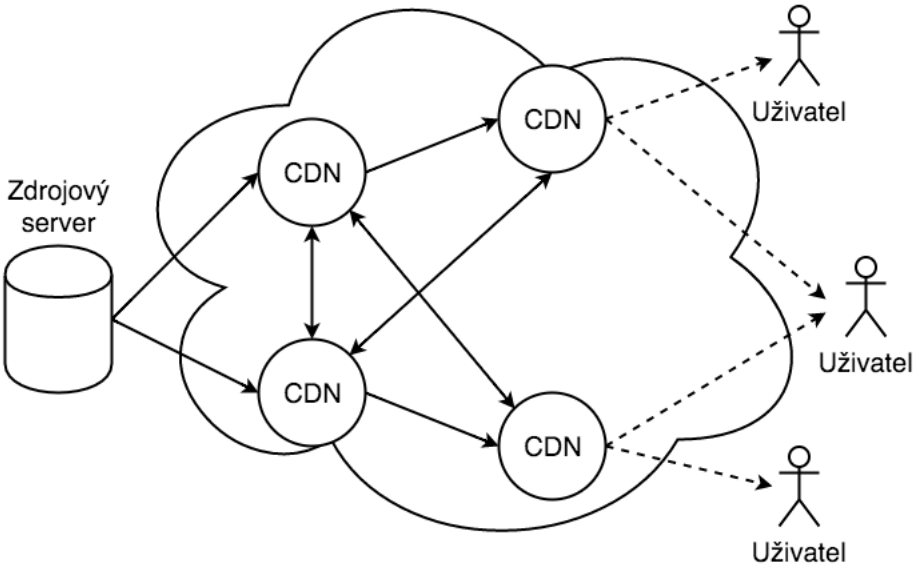
Obrázek 1: DHT

Oproti P2P sítím slouží CDN sítě k přenosu a replikaci dat co nejblíže koncovým uživatelům, na periferii internetové sítě. Jedná se především o geografickou distribuci a replikaci dat, kdy cílem je umístit repliky co nejblíže koncovým uživatelům na základě lokace. Nejčastěji jsou používané v případě globálních služeb, distribuce aktualizací operačního systému nebo příloh sociálních sítí a dalších služeb. Z jednoho zdrojového serveru jsou data přenášena do CDN tak, aby k nim měli uživatelé co nejblíže, jak z pohledu geografického (kontinent, stát), tak z pohledu síťového (minimalizace zpoždění, maximalizace rychlosti přenosu). CDN překryvná síť tak bývá obyčejně organizována jako strom, jak znázorňuje obrázek 2. Kdy jsou buď metodou pull (stahováním z centrálního serveru na CDN) nebo push (uploadem z centrálního serveru do CDN) distribuována data.

Metoda pull představuje, že pokud přijde požadavek na obsah s daným klíčem, tak server vyhledá nadřazený server u kterého požádá o data. Postup se opakuje až k serveru, který má data v lokálním úložišti nebo k centrálnímu serveru. Poprvé

stažená data se pak uchovají na hraničním serveru a to buď dočasně (cache) nebo trvale (replika).

Push metoda spočívá v tom, že zdrojový centrální server provede zápis na všechny přímo podřízené servery a ty poskytují data dále stejným způsobem. Zdrojový server pak spravuje i trvalost cache nebo replik. Doručovací strom pro CDN je opět tvořen logicky, nikoli apriori podle přenosových kapacit nebo rychlosti. Uživatelé taktéž nepracují přímo s překryvnou sítí CDN, ale přistupují pouze k jejím koncovým uzlům, které data získávají skrze privátní překryvnou síť. Optimální struktura a tvorba stromu může být objektem vědeckého zájmu.



Obrázek 2: CDN

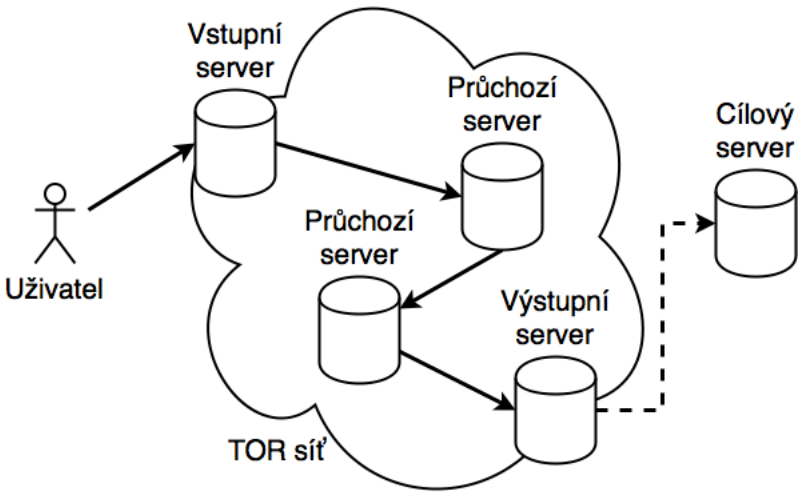
2.4.3 Bezpečnost a anonymizace

Další kategorií překryvných sítí podle použití mohou být sítě určené k zajištění bezpečnosti přenášených dat nebo anonymizace účastníků spojení. V tomto ohledu lze označit i klasické VPN a tunely (IPSec, PPTP, L2TP) za reprezentanty překryvných sítí, i když se vlastně nejedná o nový typ sítí z pohledu k přístupu k adresování, směrování, ukládání nebo distribuci dat. Tyto sítě vytváří IP nebo linkové sítě nad existující infrastrukturou IP sítí. V takto vytvořených sítích se pak používají klasické postupy a protokoly jako v běžných sítí, překryvná síť slouží jen k přemostění a zabezpečení linek nebo sítí s nedostatečnými parametry. Bezpečnost implementují pomocí použitého šifrování přenášených dat a

směrování včetně použitých protokolů se nijak neliší od protokolů použitých pro IP sítě, lze je i volně propojovat se zbylou IP infrastrukturou.

$$Data = sifra_A(hlavicka + sifra_B(hlavicka + sifra_C(payload))) \qquad (1)$$

Vhodnějším reprezentantem pro popis fungování bezpečnostních překryvných sítí je The Onion Routing protokol (TOR)[7]. Hlavní úlohou TOR je skrýt uživatele před okolním světem a zajistit anonymní přenos informací. K tomu se používá The Onion Routing protokol, stejně jako plátky cibule jsou datové pakety obalované a šifrované přes sebe (rovnice. 1). Každý server, který cestou data zpracovává, odebere jednu vrstvu šifrování, přečte si hlavičku a podle ní pošle data dále až na výstupní server, kde jsou rozšifrovaná data odeslána na cílový server (obr. 3). Celý proces anonymizace probíhá tak, že klient si z adresáře serverů vyžádá seznam uzlů. Klient následně vybere náhodnou posloupnost uzlů k zamýšlenému cíli. Přes tuto posloupnost vytvoří okruh, kde si s každým uzlem vymění klíče pro šifrování dat. Následně podle postupu, popsaného vzorcem 1, zašifruje svůj obsah a pošle do sítě. Každý z uzlů (Onion router, OR) rozšifruje příslušnou slupku a podle přečtené hlavičky posílá zprávy dále. Tímto postupem je zajištěno, že žádný z uzlů, kromě výstupního, nezná obsah dat klienta. Žádný z uzlů také nezná celou cestu v překryvné síti. Kromě posledního uzlu není nikomu znám ani cíl komunikace. Vytvořený okruh je v cibulovém směrování buď po nějakém čase nebo objemu dat přerušen a následně se zakládá jiný.



Obrázek 3: Směrování v síti TOR

Tuto základní techniku lze pro ztížení vystopování komunikujících stran rozšířit, například v rámci jednoho streamu se může využívat více disjunktních cest,

výstupní server pakety seskládá a pošle. Také je možné pořadí předávaných paketů při přenosu pozměnit - zamíchat jejich pořadím, přidávat k přeposílání náhodné zpoždění atp. Další, dnes aktuální techniku, používá síť Vuvuzela [39], která zvyšuje zabezpečení proti analýze provozu, odesílaná data maskuje náhodným šumem. Aktuálně je síť použitelná pouze pro jednoduchou textovou komunikace, protože zpoždění dosahuje až desítek vteřin. Vzhledem k tomu, že primárním cílem anonymizační překryvné sítě je anonymizace, směrování probíhá zcela záměrně náhodně, pak veškeré operace pro optimalizaci zpoždění nebo přenosové rychlosti jsou na škodu, protože by vnějšímu útočníkovi umožňovaly modifikovat směrování takovým způsobem, aby celý okruh byl sestaven přes jeho uzly. Modifikace by spočívala v jednoduchém ovlivňování stavu jednotlivých linek, tak aby bylo výhodnější poslat data jednou konkrétní cestou. Zpoždění a propustnost takových sítí je tedy horší než běžná přímá komunikace.

Dalším druhem sítí v této skupině jsou sítě, které podle jejich autorů bojují proti cenzuře, často se stávají útočištěm „kybernetických kriminálníků“. Cílem těchto sítí je opět anonymizovat obě strany komunikace a zajistit necenzurovaný přístup k datům. Tyto sítě fungují částečně jako P2P sítě pro sdílení souborů a částečně také jako cache obsahu. Jedním z rozšířených zástupců je Freenet[4].

2.5 Experimentální prostředí

Většina vlastností překryvných sítích je patrná a měřitelná pouze ve větším měřítku globálního internetu. Vytvořit si soukromé testovací prostředí nebo simulaci tak rozsáhlých systémů by nebylo vůbec jednoduché. Proto v roce 2003 vznikl PlanetLab [36][25][24], jehož členem je i CESNET [2]. PlanetLab je experimentální síť dostupných uzlů napříč všemi světadíly, tak aby pokrývala většinu Internetu. Je primárně určená pro výzkum, vývoj a testování nových internetových protokolů, distribuovaných systémů a síťových služeb. V rámci PlanetLabu aktuálně probíhá vývoj a testování následujících aplikací OceanStore[18], CoralCDN[10], projekt RON[1] (Resilient Overlay Network), projekt DHARMA[22] (Distributed Home Agent for Remote Mobile Access) a dalších.

3 Překryvné sítě a směrování dat

V této kapitole budou popsány reálné případy užití překryvných sítí a podrobně rozebrány způsoby směrování dat v nich. Jakým způsobem se překryvná síť sestavuje, jakou používá topologii, na základě jakých metrik a jak funguje směrování. Jaké jsou předpoklady a nároky pro směrování.

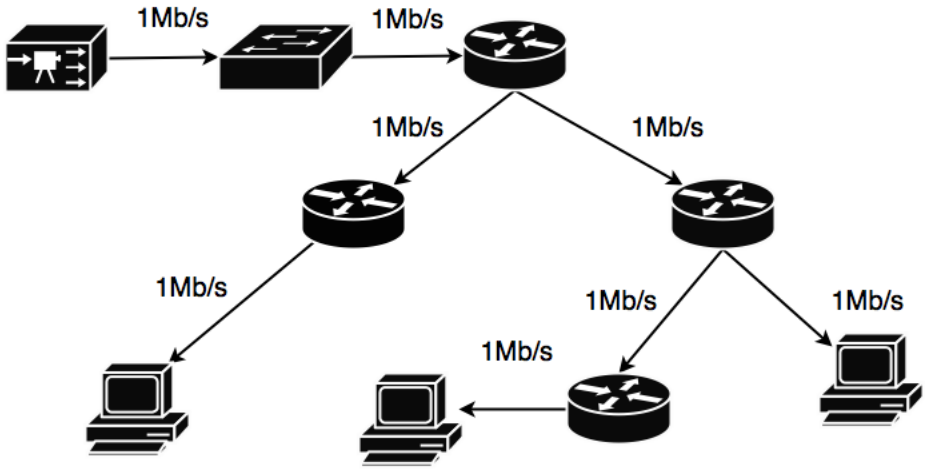
3.1 Skupinové vysílání

Z historického hlediska lze říci, že jedny z prvních překryvných sítí v internetu byly sítě implementující multicasting[21][8] nad IP protokolem. Přestože téměř všechny produkční směrovače v internetu podporovaly a podporují multicasting, tak na většině je jeho podpora vypnuta, proto skupinové vysílání v praxi funguje pouze v rámci jednoho autonomního systému (AS), kde organizace často multicasting podporují a využívají (telekonference, vysílání obsahu, IPTV atp.).

Přínosy použití skupinového vysílání pro optimalizaci přenášených dat jsou zřejmé jak ukazuje obrázek 4, data se v síti daným směrem šíří pouze jednou, ne jako v případě použití unicastu vícenásobně. Šetří se tím jak přenos dat, tak přenosové pásmo, ale i zatížení vysílacích serverů, tak směrovačů po cestě. Při použití skupinového vysílání je třeba řešit dva problémy. Jedním je udržování skupiny - seznam příjemců, kterým je vysílání adresováno, druhým je obsluha doručení dat do daných koncových uzlů. Stejně jako v případě doručování multicastu v IP sítí je třeba i v překryvné síti vytvořit doručovací strom. Strom má kořen v uzlu, který poskytuje daný proud/generuje data a postupně je vytvářena minimální kostra grafu všech uzlů skupiny.

Implementace multicastu pomocí překryvné sítě dává možnost zavést multicast i do sítí, kde není podporován. Místo použití IP směrovačů směruje překryvná síť data přes vlastní uzly. Doručovací strom je pak tvoří například pomocí hladového algoritmu pro hledání minimální kostry grafu. Překryvná síť je reprezentována úplným grafem (každý uzel je spojený s každým, mimo speciální případy a výpadky). Tvorba probíhá tak, že směrem od kořene přidáváme ty hrany, které mají nejmenší ohodnocení. Minimální kostru lze také najít obráceným způsobem a to tak, že z grafu postupně odebíráme hrany s největším ohodnocením a kontrolujeme souvislost grafu. Doručovací strom je třeba průběžně přepočítávat a přestavovat podle aktuálního stavu sítě a podle příchozích a odchozích uzlů.

Obyčejně používanou metrikou při vysílání proudu dat je propustnost dostupných linek, v případě živých přenosů nebo obousměrné komunikace (telekonference, videokonference) je klíčovým parametrem zpoždění a rozptyl. Strom je tedy tvořen jedním nebo kombinací těchto parametrů. Kromě přizpůsobování doručovacího stromu podle aktuálních parametrů sítě je možné provádět i rezervaci zdrojů.



Obrázek 4: Multicasting komunikace - IPTV

Rezervace je ovšem v překryvných sítích dosti problematická, protože po podkladové síti jsou přenášena data, která překryvné síti nenáleží, překryvná síť není schopna je ani nijak regulovat a tak nelze rezervace nijak zaručit. Na úrovni překryvné sítě lze pracovat vždy pouze s aktuální přenosovou kapacitou, kterou může překryvná síť regulovat a řídit. V tomto ohledu lze také využít predikci stavu linky na základě známého chování přenosu z krátkodobé a dlouhodobé historie.

3.2 CDN sítě

Účelem, za jakým jsou CDN sítě vytvářeny, je minimalizovat round-trip-time (RTT), time-to-live (TTL) a propustnost linek přenášejících data mezi uživatelem/klientem a zdrojem obsahu. Tento požadavek je kladen na služby nabízené klientům. Požadavky na síť jejího provozovatele jsou minimalizovat přenosy a využití linek na základě vztahů (často smluvních) mezi jednotlivými autonomními systémy, rozkládat zátěž rovnoměrně mezi dostupné uzly a efektivně replikovat poskytovaný obsah. Za přenos mezi AS se často účtují poplatky, hlavně na úrovni mezinárodních a tranzitních sítí. Cílem provozovatele CDN je tedy minimalizovat tyto náklady pomocí vhodného směrování v síti.

Z pohledu doručování obsahu jsou CDN sítě podobné skupinovému vysílání. Opět je vytvořen doručovací strom, kde kořenem stromu je původní server (zdroj obsahu) obsahující originál dat. Uzly a listy grafu jsou pak jednotlivé uzly CDN. Uzly CDN mohou být podle funkce dvojího druhu, jedny (uzly stromu) jsou pouze pro optimalizaci přenosu a druhé (listy stromu) pouze pro komunikaci s koncovými uživateli. Obě tyto funkce lze kombinovat. Jak bylo popsáno v předchozí

kapitole s úvodem do CDN, existují dva možné přístupy k distribuci dat uvnitř CDN. Jedním je push a druhým pull. Vybraný způsob je většinou dán základní architekturou CDN nebo doručovaným obsahem. Obsah může být statický, v čase neměnný (obrázky, články, videa) nebo dynamický, v čase se často měnící nebo na čase přímo závislý, např. video vysílání. O tom co a jak bude do uzlů distribuováno, rozhoduje komponenta CDN označovaná jako director. Tato komponenta slouží k řízení CDN a říká, jak bude obsah distribuován do sítě, například podle zvolených služeb zákazníka. Součástí komponenty director je kromě řízení také služba evidence zákazníků, účtování, katalog obsahu a adresář dostupných serverů a algoritmus pro řízení distribuce.

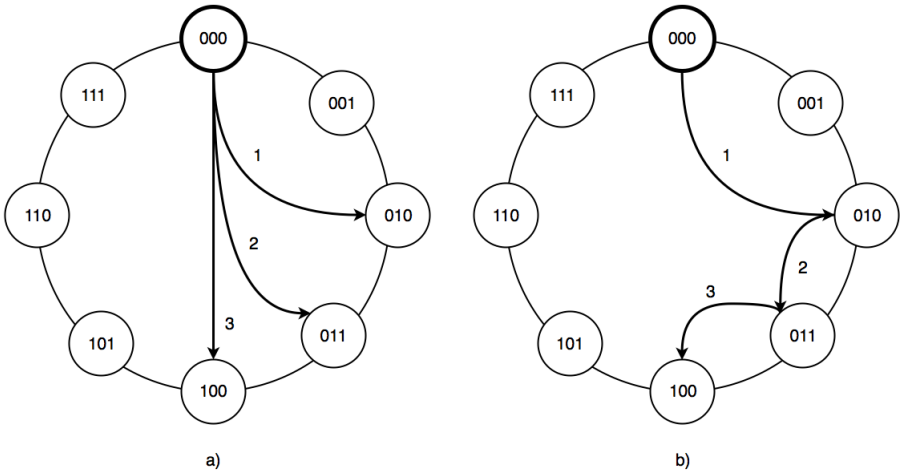
V případě distribuce živého video vysílání se síť chová podobně jako multicast, vysílání je distribuováno jen do těch uzlů, kde na jeho sledování čekají uživatelé a je tam distribuován pouze v jedné kopii, následně listy doručovacího stromu již v několika identických proudech doručují obsah přímo uživatelům. Většinou je pro to využita metoda pull, protože vysílání je nutné doručit do daného uzlu jen v případě, že má konzumenty. Zde se dynamické směrování uplatní širokou měrou, protože je třeba doručit data bez zbytečného zpoždění k uživatelům s co největší přenosovou rychlostí (kvalitou vysílání). Strom nebo jeho větev je třeba sestavit s ohledem na požadované parametry provozu, aktuální požadavky a zátěž systému.

V případě distribuce statického obsahu lze využít oba přístupy, jak pull tak push. V případě pull je vytvořena lokální replika při prvním přístupu uživatele k obsahu. V tuto chvíli je třeba vhodně zvolit server, který bude zdrojem aktuálních dat. Nejjednodušší je zvolit původní server, protože zde je zaručeno, že právě tam bude aktuální obsah dostupný. Tento přístup není vhodný s ohledem na efektivitu a jeden z cílů CDN, kterým je minimalizace zátěže zdrojových serverů a v případě nouze i obsluha požadavků z cache v případě jeho nedostupnosti. Je tedy vhodnější jako zdroj pro lokální repliku použít jiný server, který má aktuální data, server lze najít pomocí katalogu obsahu. Následně je uplatněno opět dynamické směrování tak, že je hledána taková cesta přes uzly CDN, aby byl obsah dostupný co nejdříve. S použitím dynamického směrování je možné použít ty servery, které sice nemají nejnovější repliku dat, ale cesta přes ně ke zdrojovému serveru je rychlejší. Bonusem pak může být i to, že při přenosu si mohou své lokální repliky aktualizovat i průchozí uzly.

Velkou výzvou při tvorbě CDN je také volba umístění a množství uzlů v jednotlivých lokalitách (geografická oblast, AS). Tomu většinou předchází měření a analýza návštěvníků poskytovatele obsahu. Zde je často využívána i predikce, protože množství obsahu a uživatelů průběžně roste. Dále pak jednotliví poskytovatelé často rozšiřují svoji působnost do nových regionů nebo dochází k vyššímu rozšíření internetu v dříve zaostalých regionech.

3.3 P2P sítě

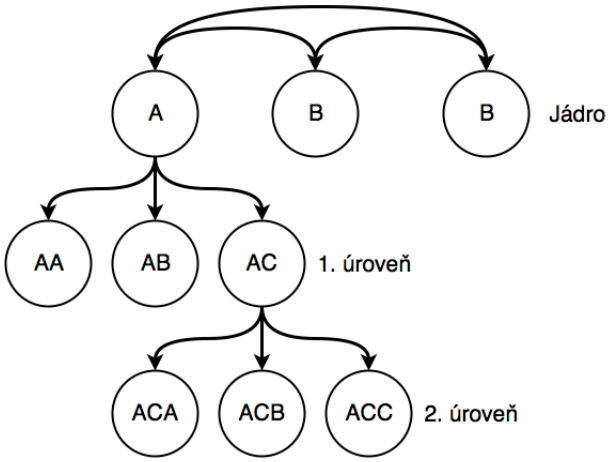
Dynamické směrování není prakticky v P2P překryvné síti příliš časté a ani obsahem aktuálního výzkumu [47], [11], [13]. Pokud P2P sítě využívají při své funkci směrování, pak se jedná většinou o hierarchické směrování. Primárně se směrování využívá ve fázi hledání dat. Jak bylo ukázáno na příkladu DHT, obr. 1 data jsou v síti reprezentována výsledkem hashovací funkce. Podle výsledného hashe jsou pak umístěna na uzly sítě nebo je na uzly sítě umístěna reference na tyto data (síť slouží pro vyhledávání dat). Směrování požadavků je pak využívané především ve fázi vyhledávání dat. V případě centralizované P2P sítě je předán seznam uzlů, které mají daná data k dispozici a následná výměna dat již probíhá přímým spojením s těmito uzly. Hierarchické P2P sítě pak používají směrování k nalezení daného uzlu v síti. Během života sítě, jak uzly do sítě vstupují a opouštějí ji, jsou zařazovány do struktury sítě a na ně je mapován obsah podle výsledků hashovací funkce. Při získávání dat je pak použit nejbližší známý uzel a dotázán na existenci dat daného klíče. Uzel odešle zpět vyhledávaná data nebo seznam nejbližších pro něj známých uzlů k danému hashi. Tento proces se pak opakuje, dokud nejsou data nalezena. Směrování je realizováno tak, že uzel místo aby vrátil seznam uzlů, které jsou k hledané hodnotě blíže, provede další dotaz sám. Tento postup opakuje další uzel dokud není konkrétní klíč nalezen a data jsou postupně vracena zpět, až doputují k uzlu, který se dotazoval.



Obrázek 5: Vyhledávání v DHT a) přímé dotazování b) směrované dotazování

Obrázek 5 zobrazuje situace, kdy uzel 000 vyhledává data s klíčem 100, nejprve je zde vidět dotaz na uzel (nebo skupinu uzlů), kterým náleží klíč 010. Vyhledávání pokračuje přes uzel 011 a v dalším kroku končí v cíli. V případě a) je dotazování přímé, nedochází k žádnému směrování, v případě b) je požadavek směrován dále na známé uzly. Z obrázku je patrné, že směrování nesleduje žádné metriky, ale

řídí se pouze klíči DHT. Obrázek 6 znázorňuje další možné uspořádání uzlů P2P sítě do stromu. Směrování dat by pak představovalo procházení tohoto stromu.



Obrázek 6: Stromová struktura DHT

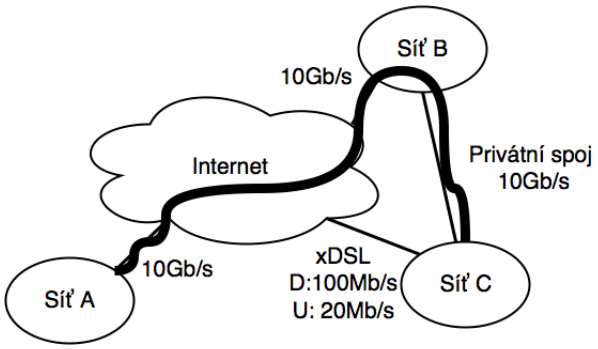
Směrování v těchto sítích je možné optimalizovat systematickou organizací dané topologie sítě. Tato systematizace může probíhat při zařazení nového uzlu nebo se může měnit v čase podle aktuálního stavu linek a uzlů. Jednou z otázek poté bude podle jakých metrik volit ohodnocení uzlů nebo skupin uzlů a jejich linek a dále jaké parametry sítě optimalizovat. V případě decentralizované P2P je třeba uvažovat, že dané uspořádání bude optimální z globálního pohledu, ale nemusí být optimální z pohledu jednotlivých uzlů a konkrétních dat, ke kterým přistupují. Dalším řešeným problémem může být znalost aktuální topologie sítě, evidence všech uzlů sítě a jejich vzájemných vztahů pro výpočet optima.

Dynamické směrování je možné do P2P zavést v případě, že jednotlivé klíče DHT spravuje fyzicky více uzlů, pak budeme v každém kroku vybírat ten uzel, který splňuje nejlépe definované parametry pro zvolený provoz. Rozhodování o vhodné trase muže být prováděno vždy v každém uzlu samostatně nebo může být řízeno hlubší znalostí topologie a jednotlivých metrik použitých spojů. Například v každém kroku vyhledávání může být dotazujícímu uzlu odeslán seznam s nejbližšími uzly a parametry dostupných linek. Dotazující pak spočítá optimální trasu pro průchod celou cestou k cíli. Inicializace výpočtu bude zatížena nutnou režií pro získání těchto informací, ale následný přenos dat bude optimální z hlediska stanovených metrik.

3.4 Zajištění kvality služeb

Překryvné sítě pro zajištění kvality služeb jsou aktuálně reprezentovány sítěmi Service Overlay Network (SON)[12] a Resilient Overlay Netwok (RON)[1].

Resilient Overlay Network je navržena s cílem minimalizovat dopady náhlých změn na sítích, především výpadky spojů (rychlejší změna trasy, než ji provede BGP nebo jiný internetový směrovací protokol), přetížení nebo zahlcení linek (útoky nebo anomálie způsobená provozem na síti kvůli nějaké významné události). Uzly RON neustále sledují stavy jednotlivých linek mezi uzly sítě a bez prodlení na ně reagují a provoz přesměrovávají. Platforma pro síť je vytvořená jako externí knihovna, kterou je možné zakomponovat do existujících programů a která následně zajistí přenos sítí, tak aby byl optimální podle definovaných potřeb. Logika v knihovně rozhodne, jestli je vhodnější použít přímé spojení pomocí klasických linek nebo přemostit přenos pomocí překryvné sítě. Schéma fungování znázorňuje obrázek 7. RON využívá pro rozhodování základní metriky, rychlost, spolehlivost a zpoždění. Knihovna je navržena pro spolupráci s aplikacemi, ke kterým je připojena. Je možné specifikovat i vlastní aplikační metriky. Výsledná metrika a politika použitá při směrování je pak založena na explicitních pravidlech, podle zdroje, cíle a třídy přenosu. Přenášená data jsou pak označena nejen zdrojem a cílem, ale také třídou provozu a označením použité politiky. Definice jednotlivých politik může směrování dále ovlivňovat, například zakázat přenos po vybraném spoji. Každý uzel sítě se skládá z několika částí, první je předavač, který na základě vyhodnocení směrování předává data dále, výkonnostní databáze, která obsahuje získané hodnoty jednotlivých metrik a jsou z ní následně odvozeny možné cesty a databáze politik, která dále rozhoduje, která cesta může být pro přenos použita.



Obrázek 7: Koncept fungování RON

Hlavním cílem RON je minimalizovat problémy v internetu způsobené především výpadky linek a nízkou přenosovou rychlostí způsobnou, ať již nízkou kapacitou linek nebo jejím aktuálním zahlcením. Oproti tomu Service Overlay Netwok se snaží zcela koncepčně zajistit konkrétní parametry komunikace s důrazem především na kvalitu služeb (QoS) pro aplikace jako například VoIP, video na požádání a další online aplikace, kde je třeba garantovat definovanou minimální kvalitu přenosu.

QoS jak v překryvných tak běžných sítí předpokládá prioritizaci některého přenosu nad ostatními. Mohou být definovány třídy přenosu podle jejich požadavků. Požadavky na přenos pak mohou být pomocí prioritních front nebo pomocí token bucket algoritmu. Často se také používá rezervace přenosového pásma podle nároků aplikace definovaných při zahájení přenosu. Použití rezervací je ale v případě překryvných sítí nanejvýše problematické, protože překryvná síť nemá dostatek informací o celkové šířce dostupné pro použité spoje, ale může je pouze odhadovat na základě krátkodobých nebo dlouhodobých měření. Překryvná síť ani nedokáže spolehlivě požadovanou kapacitu rezervovat, protože linky mohou být kdykoli zatížené přenosem mimo překryvnou síť, který na linkách také probíhá paralelně. Lze tedy pracovat pouze s pravděpodobností schopnosti garantovat dané parametry přenosu. Ke zvýšení pravděpodobnosti mohou přispět metody predikce zatížení a dostupné kapacity linky, která přenos realizuje. Tomuto tématu, jak predikovat parametry linky v blízké budoucnosti se budou věnovat další kapitoly práce.

3.5 Anonymizační sítě

Anonymizační sítě jako jsou TOR[7], ORTA[37] nebo Vuvuzela[39] využívají dynamické směrování k anonymizaci účastníků a ochraně proti odposlechu neustálou změnou tras. Sledovanou metrikou je pravděpodobnostní rozdělení generátoru náhodných cest, aby byla zvolená cesta dostatečně náhodná a tím i bezpečná proti odposlechu nebo sledování. Cesty musí být volené tak, aby splňovaly požadavek na minimální počet skoků a aby byly uzly a jejich pořadí volené s rovnoměrným rozdělením pravděpodobnosti. Pokud by docházelo k jakémukoli deterministickému směrování na základě libovolných metrik, pak by bylo možné vkládat do sítě uzly tak, aby s pravděpodobností p vedla cesta, právě přes tyto vložené uzly. S možností vložit větší počet uzlů by pravděpodobnost, že provoz půjde přes vložené uzly, rostla. Všechny sítě fungující na tomto principu musí v návrhu usilovat o to aby poměr vlastních a vložených uzlů musel být co největší, aby nebylo možné kompromitovat síť tak snadno nebo to bylo finančně velice náročné.

3.6 Shrnutí směrování v překryvných sítích

V každém druhu překryvných sítí popsaných předchozích sekcích probíhá směrování různými způsoby. Některé sítě jsou strukturované, tvoří například prostor dat (2D plocha, n-rozměrná kostka atp.) rozdělený na menší oblasti a směrování probíhá mezi těmito oblastmi. Jiné strukturované sítě tvoří například doručovací stromy, hvězdu, kruh a nebo kombinace. Pokud není struktura sítě pevně dána, pak se tvoří zvolené/definované struktury s ohledem na splnění de-

finovaných cílů, rychlost, spolehlivost doručení, maximální logickou vzdálenost a dostupné zdroje. Aby bylo možné strukturu tímto způsobem sestavit, tak je třeba znát vlastnosti sítě. Vlastnosti dané sítě vychází z prostředků, které jsou jí dostupné skrze podkladovou síť.

Tvorba topologie a její uspořádání je tedy dáno všemi požadavky, které jsou na danou síť kladeny a aktuálními a dostupnými informacemi o stavu sítě a parametry linek.

4 Směrovací metriky

Pro směrování v překryvných sítích se používají dvě základní metriky, jednou je dostupná přenosová rychlost, druhou je zpoždění. Další sledované metriky pak mohou být spolehlivost linky (ztrátovost paketů, chybovost při odesílání/příjmu), stabilita (kolísání) přenosových parametrů linky nebo její opakované přerušení spojení. Každý z těchto parametrů je možné měřit a následně ho využít při směrování požadavků sítí.

Zásadní komplikací překryvných sítí je v získávání těchto dat, žádná nebo pouze malá znalost nižší vrstvy, kterou pro svůj provoz využívá a také v mnoha případech i fakt, že uzly překryvné sítě tvoří převážně koncové uzly, uživatelské stanice, servery atp. V současné době se také rozvíjí řízení přenosu (traffic engineering, TE) nebo dynamické řízení přenosu, které může překryvným sítím komplikovat situaci, tím, že se dynamicky mění směrování v podkladové síti a tím i parametry, které může překryvná síť sledovat. Na druhou stranu postupům řízení provozu na nižších vrstvách mohou překryvné sítě také narušovat nebo ovlivňovat funkčnost.

Aktuální vědecké zkoumání se tak zaměřuje jak na aplikaci principů známých z překryvných sítí do řízení provozu např. IP sítí, tak v rámci softwarově definovaných sítí [17][16]. Řada publikací se zabývá interferencí řízení síťového provozu a překryvných sítí, jejich možnou spoluprací, případně řízením jedné sítě druhou. Publikované práce popisují, že čisté řízení podkladové sítě podle požadavků a pravidel překryvné sítě vede k nestabilitě spojení a linek.

4.1 Sledování a měření parametrů v překryvných sítích

Problematika sledování a měření parametrů překryvných sítí je oproti podkladovým, základním, sítím specifická tím, že nejsou dostupné konkrétní informace o dostupných prostředcích, ani konkrétní konfiguraci podkladové sítě. Překryvná síť nemá informaci o tom, jaké jsou priority a metriky podkladové sítě a jaký je použitý směrovací protokol. Není známo jestli podkladová síť využívá distance

vector nebo link state směrování. Nemáme ani dostupné informace o tom, jak je síť rozlehlá a z jakých technologií se skládá. Měřené hodnoty jsou také zatížené, resp. ovlivněné ostatním okolním provozem, který s vytvořenou překryvnou sítí interferuje.

Měření základních metrik použitelných pro směrování tak může probíhat dvojím způsobem, jedním je aktivní měření, kdy jsou mezi uzly překryvné sítě posílány zprávy, pomocí kterých je možné dané metriky získat. V případě detekce zpoždění na lince je možné posílat krátké zprávy a měřit čas mezi odesláním a příjmem. Získaný čas je časem cesty zprávy k cíli a odpovědi zpět, RTT, round trip time. S využitím znalosti Christianova algoritmu pro synchronizaci času můžeme odvodit dobu cesty paketu k cíli. Pro získání dostupné přenosové rychlosti jsou vysílány delší zprávy, kde se projeví dostupná přenosová rychlost víc, než u krátkých, kterou jsou zatížené spíše zpožděním. Pokud bude zpráva dostatečně dlouhá, pak bude možné zpoždění linky zanedbat.

Aktivní měření parametrů linky ovšem generuje zbytečnou zátěž sítě a může tak negativně ovlivňovat i vlastnosti překryvné sítě. V případě, kdy je překryvná síť reprezentována jako úplný graf na všech uzlech sítě, je třeba každé sledování provést alespoň N^2. Zasílání zpráv je vhodné provádět v obou směrech dvou komunikujících uzlů, protože je tam možné odhalit i asynchronní linky. Existují algoritmy a postupy, kdy se počet měření redukuje na nutná měření a další hodnoty jsou odvozeny pomocí aproximace [44].

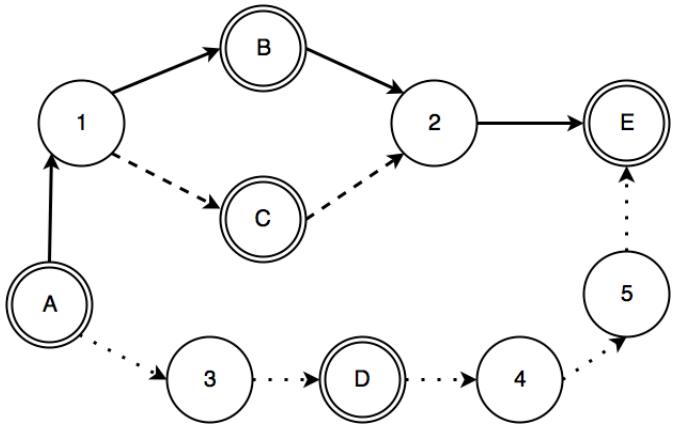
Vhodnějším postupem než aktivním měřením je pasivní zjišťování těchto vlastností z procházejících dat. Základní znalost o latenci můžeme získat inspekcí TCP hlaviček, které obsahují RTT. Údaj o rychlosti přenosu je možné zjistit z procházejících dat a doplněním provozních informací do hlaviček překryvné sítě.

4.2 Další sledovatelné parametry

Dalšími sledovatelnými parametry ovlivňujícími efektivitu přenosu a směrování přes překryvné linky mohou být kromě měřených hodnot daných linek i další vlastnosti komponent překryvné sítě, například zatížení výpočetních prostředků jednotlivých uzlů, interference jednotlivých linek mezi sebou, příslušnost uzlů k jednotlivým AS.

Obrázek 8 znázorňuje překryvnou síť, uzly A, B, C, D, E jsou uzly překryvné sítě, uzly 1, 2, 3, 4, 5 jsou uzly, přes které komunikace prochází, ale nejsou součástí překryvné sítě, například směrovače. Spojení mezi uzly A a E lze realizovat třemi různými cestami, jedna je označena plnými spoji (A1B2E), druhá přerušovanou (A1C2E) a třetí tečkovanou čárou (A3D45E). Z pohledu překryvné sítě jsou čísly uzly neviditelné. Pokud zvolíme první cestu jako primární a další dvě jako záložní, pak pokud dojde k výpadku nebo zahlcení na spoji A1 nebo 2E, pak to ovlivní

jak první tak druhou cestu a bude muset být použita cesta třetí. Překryvná síť obecně nemá znalost podkladové topologie, v rámci směrování rozlišuje pouze vlastní uzly (označené písmeny). Překryvná síť chápe cesty A1B a A1C jako dvě různé cesty AB a AC. V praxi může pak dojít k tomu, že cesta A1B2E bude nahrazena cestou A1C2E, na které se zjištěné parametry nebo událost objeví také a bude třeba použít až cestu A3D45E. Další situace, která může nastat v případě, kdyby překryvná síť poskytovala agregaci vybraných linek, tak v případě agregace na linkách A1B2E a A1C2E nemusí dojít k žádnému zisku, protože obě cesty využívají společných uzlů a jejich spojů, konkrétně A1 a 2E kde by rozdělená zátěž znovu sčítala.



Obrázek 8: Překryvná síť se společnými linkami

Často je překryvná síť tvořena jak specializovanými uzly (směrovače, servery určené pro poskytování služeb překryvné sítě), tak běžnými uživatelskými stanicemi (dle nastavení, jak poskytují služby, tak je od jiných využívají). Běžný uživatelský přístup je vůči službám převážně konzumní, lze tedy předpokládat, že nastavení těchto uzlů bude asymetrické, budou více čerpat a méně nabízet. Kvalitu služeb nabízených těmi uzly dále ovlivní mnohem více aktuální zatížení jejich systému, dostupné výpočetní prostředky (rychlost výpočtů, rychlost předávání zpráv, rychlost šifrování atp.), tak dostupné přenosové pásmo, bude třeba uspokojit jak požadavky překryvné sítě, tak požadavky lokálních aplikací. Specializované uzly jsou také ovlivňovány danými vlastnostmi, ale v mnohem menší míře. Při efektivním směrování dat v překryvné síti je vhodné brát v úvahu i tyto vlastnosti jednotlivých uzlů a zahrnout je například jako aditivní složku při výpočtu cest v síti.

5 Predikce stavu linek a sítě

Pokud jsme schopni měřit a vyhodnocovat vlastnosti, stav a chování linek v překryvné síti, pak se nabízí otázka možnosti předpovídat zatížení sítě a tím optimalizovat směrování. Obyčejně je změna směrovacích tabulek vyvolána zásadní změnou na síti, výpadkem linky, výpadkem routeru, zásadní změnou parametrů dané linky nebo zásahem administrátora. Tento reaktivní postup může být časově relativně náročný, zvláště pak čas konvergence, než se síť po této změně opět stabilizuje.

Pokud bychom k reaktivnímu přístupu řízení sítí připojili i proaktivní změny směrování, pak by bylo možné i přes změny v podkladové síti, kdy například dojde k saturaci nebo zahlcení některé podkladové linky, kterou jsme schopni predikovat, tak pro zachování kvality služby včas přesměrovat provoz po jiných linkách, kde je dostatečně volná kapacita.

5.1 Časové řady a predikce

Síťový provoz představuje časovou řadu, která má klasické statistické vlastnosti jako střední hodnotu, směrodatnou odchylku a další. Časová řada se skládá z jednotlivých hodnot měření s konstantním časovým odstupem jednotlivých měření. Pokud není řada kompletní nebo časové údaje chybí, pak je třeba hodnoty interpolovat tak, aby byl odstup jednotlivých měření konstantní. Časová řada síťového provozu může být tvořena z různých hodnot, které nám definují použitou metriku při směrování, například aktuální přenos na lince, zpoždění doručení zpráv, obousměrné zpoždění, spolehlivost, ztrátovost a další. Z předchozích naměřených hodnot je pak možné predikovat pouze jednu následující hodnotu nebo skupinu nových hodnot.

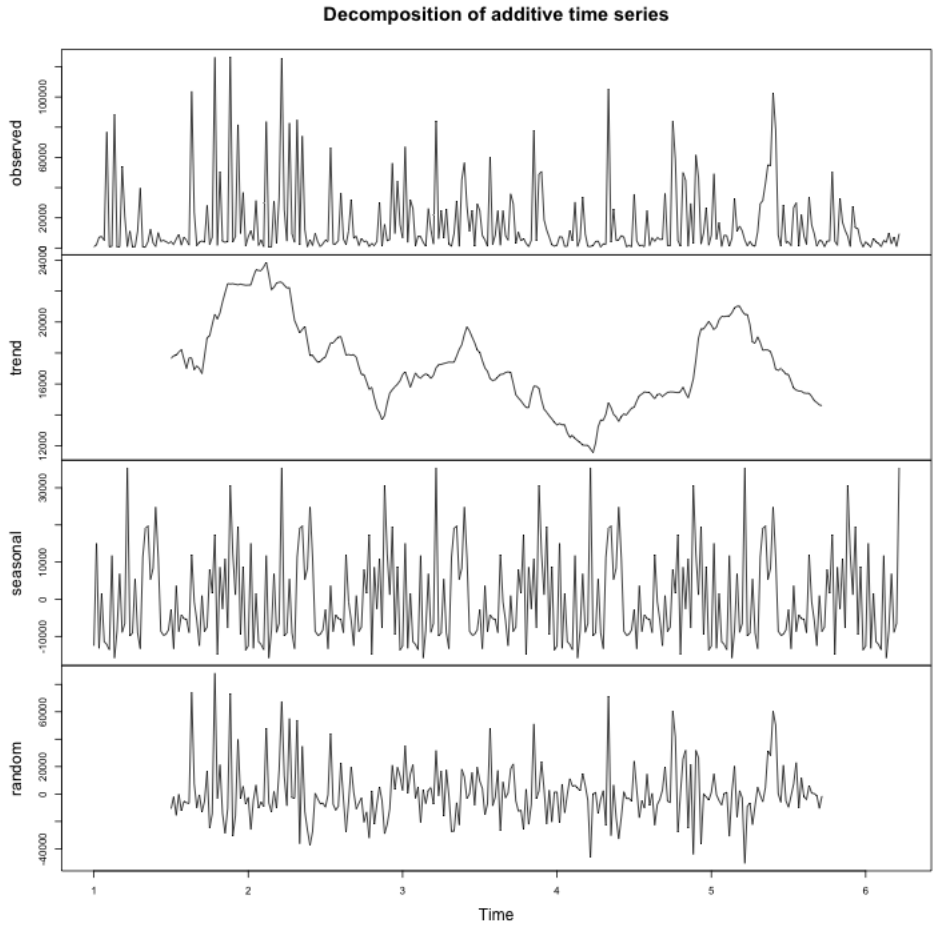
Časové řady je možné zkoumat s ohledem na jejich vlastnosti. Jedním z dělení časových řad může být, jestli je časová řada stabilní nebo nestabilní. Pokud je časová řada stabilní, pak její vlastnosti nezáleží na čase, kdy ji sledujeme (má konstantní statistické vlastnosti). Oproti tomu nestabilní časové řady obsahují složky, které jsou proměnné v čase, například trend nebo periodu. Časová řada je kombinací trendové složky, sezónní složky a zbytkové (chybové složky), vztah vyjadřuje rovnice 2. V publikacích se uvádí ještě cyklická složka. Následující obrázky predikce jsou výstupem z jazyka R [30] a dostupných knihoven pro práci s časovými řadami. Jazyk R je určen pro zpracování statistických dat a jejich zobrazení.

$$Y_t = T_t + S_t + e_t \tag{2}$$

Časová řada síťového provozu je klasickým případem proměnlivé časové řady, protože zcela jistě vykazuje periodické chování, které je dáno uživateli, během

noci je provoz nižší než během dne, kdy jsou uživatelé aktivní. Časové řady síťového provozu v dlouhodobém měřítku vykazují jasný vzrůstající trend, aplikace přenášejí větší množství dat a počet uživatelů a aplikací postupně roste. Příkladem stabilní časové řady pak může být například bílý šum, který neobsahuje žádnou periodickou ani trendovou složku a má statisticky konstantní vlastnosti. Nestabilita časové řady síťového provozu je pro predikci pozitivní vlastnost, protože základní predikci lze dělat na základě periodické a trendové složky. Z obrázku 9 je patrné rozložení časové řady na složky, můžeme vidět složku trendu, sezónní (periodickou) a náhodnou.

V praxi se před predikcí používá ještě předzpracování, kdy se časová řada upraví, například se ořežou extrémy, řada se linearizuje, interpolují se chybějící hodnoty, získané hodnoty se vyhladí (filtrují), normují se na rozsah $< 0; 1 >$ nebo $< -1; 1 >$ a další. Následně jsou data použita pro samotnou predikci.



Obrázek 9: Dekompozice časové řady

Pro predikci časových řad se dnes používá několik základních technik, metoda jednoduché lineární regrese, auto-regrese (AR), klouzavého průměru (MA), kombinace předchozích (ARMA, ARIMA, FARIMA), metody postavené na dekompozici (STL), fraktálová geometrie a neuronové sítě. V následujících kapitolách budou vybrané metody popsány a shrnuty jejich vlastnosti.

Z vlastností internetového síťového provozu popsaného v článcích [38][6][19][43] je síťový provoz soběpodobný (termín z oblasti fraktálové geometrie, kdy se jednotlivé části v různých měřících navzájem podobají) a nelineární. Této vlastnosti různých jevů je využíváno i v řadě dalších vědních oborů, elektronice, ekonomii, fyzice, chemii a dalších, kde je třeba predikovat stav jevu těchto vlastností. Různé vědní obory k této problematice přistupují různě, resp. specializují se na různé metody predikce[15]. V informačních technologiích převládá použití neuronových sítí, v přírodních vědách se využívá teorie chaosu a fraktálová geometrie nebo jednodušší metody z oblasti filtrace a predikce, kde je ovšem značnou nevýhodou potřeba hlubší znalosti predikované časové řady, obecně záleží na požadované přesnosti a zvoleném aparátu nebo hloubce znalosti predikovaných dat.

5.2 Predikce pomocí modelů ARMA

ARMA model je kombinací autoregresního (AR) modelu a klouzavého průměru (MA). Auto-regresivní model $AR(p)$ odvozuje predikované hodnoty z hodnot předchozích jako jejich lineární kombinaci. V rovnici 3 je y_t predikovaná hodnota y v čase t z předchozích p hodnot pomocí vektoru ϕ. Hodnota c představuje konstantu a e_t symbolizuje náhodnou složku, která má nulovou střední hodnotu a normální distribuci. Tyto modely se hodí převážně pro stacionární časové řady. Nestacionární časové řady lze na stacionární převést pomocí diference časové řady (rovnice 7).

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + e_t \tag{3}$$

Model klouzavého průměru $MA(q)$ je pak definován jako lineární kombinace chyb jednotlivých předpovědí 4.

$$y_t = c + e_t + \Theta_1 e_{t-1} + \Theta_1 e_{t-1} + ... + \Theta_q e_{t-q} \tag{4}$$

Kombinací auto-regrese a klouzavého průměru pak získáváme $ARMA(p, q)$ 5 model, který lze dále rozšiřovat na $ARIMA(p, d, q)$ 6 nebo $FARIMA(p, d, q)$ 6 nebo jejich další varianty, například s uvažováním sezónní složky $SARIMA(p, d, q)(P, D, Q)$.

$$y_t = c + e_t + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=0}^{q} \Theta_j e_{t-j} \tag{5}$$

Pokud systém v dalších stupních diference časové řady (rovnice 7 a 8) vykazuje stacionární chování, pak na tento stupeň můžeme aplikovat $ARMA(p, q)$ model a vzniká tím $ARIMA(p, d, q)$. To je proces, kde je d úroveň diference, p je autoregresivní stupeň a q stupeň pohyblivého průměru, p a q nabývají nezáporných celočíselných hodnot. Dále lze odvodit proces $FARIMA(p, d, q)$, což je proces, který je založen na ARIMA procesu a oba lze vyjádřit s pomocí operátoru zpětného posunutí B následujícím vztahem 6. V případě $ARIMA(p, d, q)$ jsou všechny parametry celočíselné a nezáporné, pro $FARIMA(p, d, q)$ je parametr d reálný.

$$\phi_p(B)(1 - B)^d Y_t = \Theta_q B e_t \tag{6}$$

$$(1 - B)^1 Y_t = \Delta Y_t = Y_t - Y_{t-1} \tag{7}$$

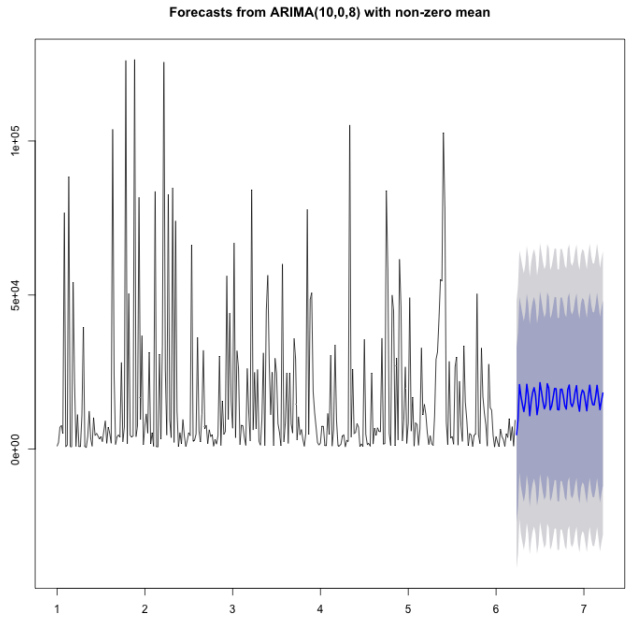
$$\begin{aligned} \Delta^2 Y_t = \Delta(Y_t - Y_{t-1}) = \\ \Delta Y_t - \Delta Y_{t-1} = \\ Y_t - Y_{t-1} - (Y_{t-1} - Y_{t-2}) = \\ Y_t - 2Y_{t-1} + Y_{t-2} \end{aligned} \tag{8}$$

V rovnicích $\{Y_t : ..., -1, 0, 1, ...\}$ představuje časovou řadu, d představuje diferenci časové řady (rovnice 7 a 8), $\{e_t : ..., -1, 0, 1, ...\}$ představuje bílý šum s nulovou střední hodnotou a parametry p a q odpovídají parametrům $AR(p)$ a $MA(q)$.

Příklad jednoduché predikce pomocí modelu $ARIMA(p, d, q)$ s parametry $p = 10, d = 0, q = 8$ je znázorněn na obrázku 10.

5.3 Predikce pomocí neuronových sítí

V informatice jsou pro predikci časových řad často používány neuronové sítě, neuronových sítí existuje celá řada. Neuronové sítě se skládají z řady výpočetních jednotek (neuronů), které jsou vzájemně propojeny ohodnocenými hranami. Neurony jsou organizované do vrstev, základní je vstupní vrstva, která obsahuje vstupní neurony odpovídající velikosti vstupu, následuje několik skrytých vrstev a poté výstupní vrstva opět odpovídající velikosti výstupu. Jednotlivé druhy

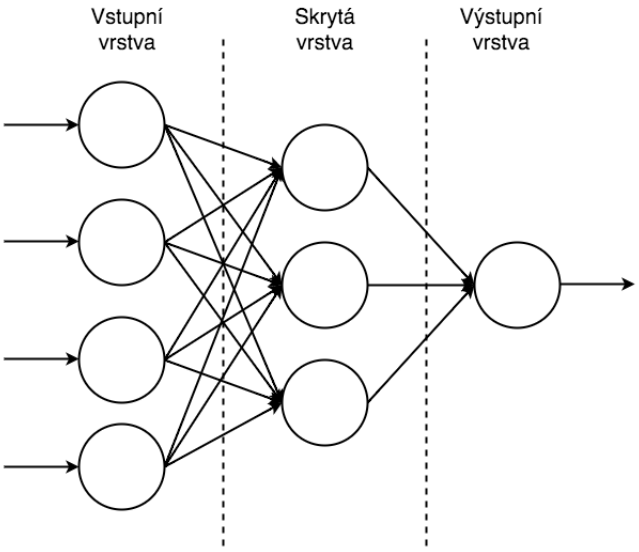


Obrázek 10: Predikce pomocí ARIMA(10,0,8)

neuronových sítí se pak liší počtem organizací těchto vrstev (se zpětnou propagací obsahují zpětné vazby). Základní organizace neuronových sítí je znázorněna na obrázku 11.

Neuronová síť je schopna postupným učením na trénovacím vzorku přizpůsobit ohodnocení jednotlivých hran, čímž se adaptuje na funkci, která je ukryta v datech a která tyto data charakterizuje. Výhodou využití neuronových sítí je právě její schopnost samo učení z naměřených dat, kdy nepotřebujeme znát přesné parametry a charakteristiky provozu. Neuronové sítě dokáží díky natrénovaným datům z minulosti predikovat budoucnost na základě „odhalení“ vnitřní funkce charakterizující vztah mezi aktuálními hodnotami a jejich projevem v budoucnosti. Správně natrénovaná neuronová síť je schopna reagovat i na data mimo cvičnou množinu, proti tomu špatně natrénovaná síť může být schopna správně zpracovávat pouze trénovací vzorky. Záleží tedy i na kvalitě dat pro trénování neuronové sítě a následné ověření jejích kvalit.

Neuronových sítí existuje celá řada, jednotlivé sítě se od sebe liší svojí topologií, způsobem učení, zprostředkováním zpětné vazby, dočasnou pamětí a podobně. Ne všechny modely neuronových sítí mají stejnou schopnost řešit různé problémy. Neuronové sítě využitelné k rozpoznávání vzorů v obrazech nemají stejnou schopnost predikce časových řad. Kromě schopnosti přesné predikce jsou neuronové sítě hodnoceny dalšími parametry jako například rychlost učení, schopnost zlepšit pre-



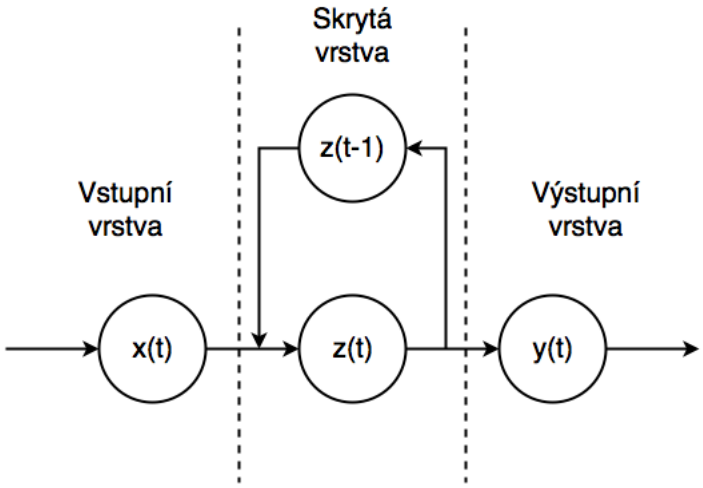
Obrázek 11: Vícevrstvá neuronová síť

dikci vhodnou úpravou a podobně. Pro predikci soběpodobného síťového provozu lze použít některou z následujících neuronových sítí, vícevrstvá perceptronnová síť, Elmanova síť[14][41], NARX síť, LTSM síť, obecná rekurentní síť a další.

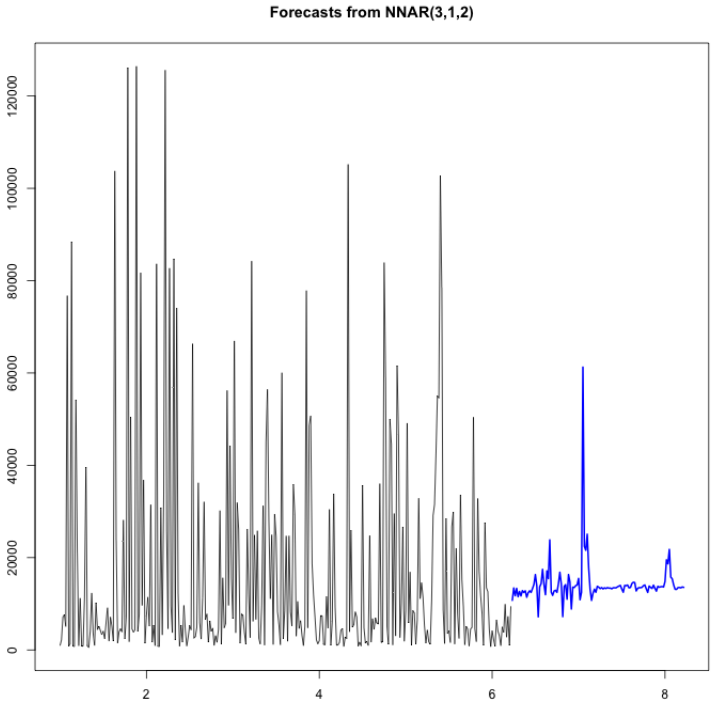
Pro příklad použití neuronových sítí je vybrána Elmanova rekurentní neuronová síť. Elmanova síť obsahuje dvě vrstvy s algoritmem učení pomocí zpětné propagace. Skrytá vrstva má zpětnou vazbu skrz stavovou vrstvu na vstupní vrstvu. V každém časovém okamžiku je možné výstup ze skryté vrstvy použít jako vstup do stavové vrstvy a ta je v dalším kroku použita jako součást vstupu do celé sítě. Struktura Elmanovi sítě je zobrazena na obrázku 12. Množství neuronů ve vnitřní i stavové vrstvě je shodné, váhy spojů mezi nimi mají hodnotu 1.

Výhodou neuronových sítí je jejich schopnost generalizace, odhad složitých jevů a samo korekce. Určitou nevýhodou, zvláště s orientací na koncová zařízení, jsou relativně vysoké požadavky na výpočetní výkon a paměťový prostor při trénování sítě. Po nastavení vah neuronů v síti je následné využití relativně nenáročné. Síť je možné trénovat i průběžně pomocí korekce s novými daty.

Na obrázku 13 je znázorněna predikce časové řady, která je stejná jako na obrázku 10, ovšem predikovaná pomocí jednoduché dopředné neuronové sítě s využitím funkce *nnetar* z balíku *forecast* jazyka R. Funkce *nnetar* implementuje jednoduchou dopřednou neuronovou síť s jednou skrytou vrstvou. Použitá neuronová síť má 4 vstupní neurony, 2 skryté a 1 výstupní, predikuje tedy pouze jednu hodnotu. Predikce dalších hodnot je zařízena rekurzivním voláním s použitím predikovaných hodnot.



Obrázek 12: Elmanova neuronová síť



Obrázek 13: Predikce pomocí NNAR(3,1,2)

5.4 Predikce s využitím teorie chaosu

S vývojem teorie chaosu se zkoumají i možnosti predikce pomocí této teorie. Metody teorie chaosu na predikci síťového provozu jsou aktuálně zkoumané téma podle publikací [20], [15], [26], [40], [9]. Síťový provoz lze predikovat pomocí teorie chaosu s využitím Ljapunovova exponentu. Pomocí teorie chaosu lze obecně predikovat chování nelineárních systémů, které mají nějaký skrytý řád, avšak tento řád není na první pohled viditelný a jeví se jako náhodný systém.

Chaotické časové řady je možné znázornit pomocí D dimenzionálního abstraktního prostoru stavů zvaného fázový prostor 9. V něm každá osa představuje jednu dimenzi stavů a čas je zde implicitní, volíme ovšem zpoždění T jako v předchozích případech, které udává z kolika předchozích hodnot počítáme předpověď. Po určitém čase vyvíjení systému vznikne ve fázovém prostoru křivka, tato křivka po dostatečně dlouhé době začne zvýrazňovat strukturu, které se říká atraktor[5]. Atraktor představuje konečný stav sledovaného systému, obyčejně bývá fraktálem.

$$\begin{aligned} Y(I), I \in [1[N-(D-1)T]] : \\ Y(1) = [x(1), x(1+T), ..., x(1+(D-1)T)] \\ Y(2) = [x(2), x(2+T), ..., x(2+(D-1)T)] \\ Y(3) = [x(3), x(3+T), ..., x(3+(D-1)T)] \\ ... \\ Y(I) = [x(I), x(I+T), ..., x(I+(D-1)T)] \end{aligned} \tag{9}$$

Chaotické atraktory mají velkou citlivost na vstupní podmínky. Chaotické systémy charakterizuje Ljapunův exponent, který určuje stupeň chaotičnosti daného systému. Ljapunův exponent vyjadřuje, zda blízké dráhy konvergují nebo divergují. Pro každou dimenzi systému existuje právě jeden Ljapunův exponent. Pro charakteristiku systému je nejdůležitější ten největší, ovlivňuje dlouhodobé chování systému.

Jestliže je Ljapunův exponent záporný, dráhy v čase konvergují, takový dynamický systém není citlivý vůči počátečním podmínkám. V případě, že je daný exponent kladný, pak dráhy atraktoru mezi sebou divergují a daný systém je citlivý na počáteční podmínky. U chaotického systému se musí alespoň v jednom případě trajektorie drah atraktoru exponenciálně vzdalovat (musí divergovat), tedy musí mít alespoň jeden Ljapunův exponent kladný.

Jeden způsob jak hodnotu exponentu získat, je zvolit si několik blízkých bodů, které necháme v čase rozvíjet a přitom sledujeme rychlost růstu jejich vzájemné vzdálenosti. Tento postup se nazývá Wolfův algoritmus.

Pomocí Ljapunovova exponentu lze určit horizont predikce, tedy maximum budoucích bodů, které je možné predikovat. Tento horizont lze vypočítat pomocí rovnice 10.

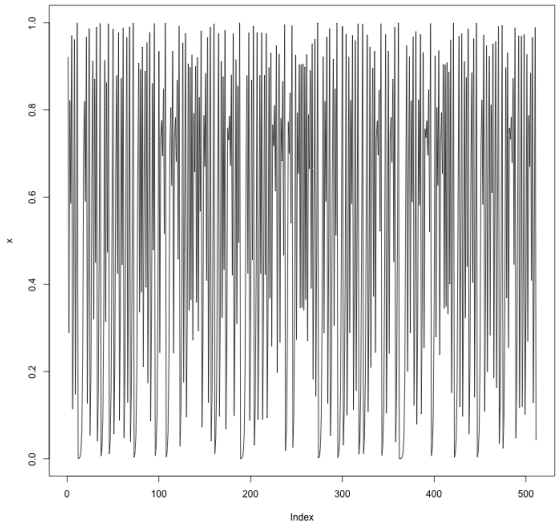
$$T_{max} = \frac{1}{\lambda_{max}} \ln \frac{\Delta}{\delta_0} \tag{10}$$

Kde λ_{max} je největší Ljapunův exponent, Δ je požadovaná maximální chyba a δ_0 je neurčitost v měření počátečních podmínek. Ze vztahu je patrné, že ideální hodnota λ_{max} je blízká nule, získáme delší horizont predikce. Z prací zkoumajících vlastnosti síťového provozu ovšem plyne, že Ljapunův exponent je v praxi relativně vysoký a tak je možné predikovat pouze blízkou budoucnost. Nevýhodou predikce pomocí teorie chaosu s využitím Ljapunova exponentu může být náročný výpočet Ljapunova exponentu a již zmíněná citlivost na počáteční podmínky.

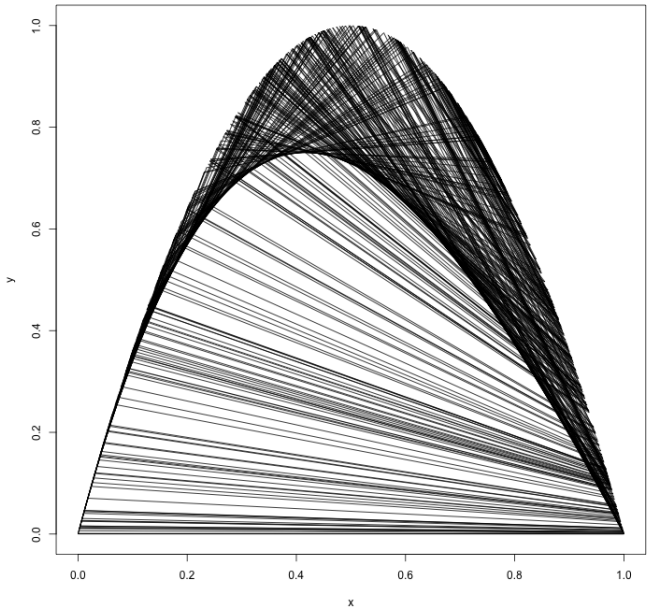
Jako příklad lze uvést algoritmus prezentujíc použití tohoto postupu a Ljapunova exponentu [49]. Nejprve zvolíme časovou řadu jako předpis 11 a dvoudimenzionální stavový prostor.

$$\begin{aligned} x(0) &= 0.36 \\ x(t+1) &= 4 * x(t) * (1 - x(t)) \end{aligned} \tag{11}$$

Použitá časová řada je znázorněna na obrázku 14 a ve stavovém prostoru pak 15.



Obrázek 14: Časová řada dle definice 11



Obrázek 15: Časová řada 11 ve 2D prostoru podle 9

Novou predikovanou hodnotu pak hledáme pomocí existujících křivek a nejkratších vzdáleností. Známe pozici posledního bodu fázového prostoru a hledáme nejbližší bod na dalších křivkách v okolí, získáme vzdálenost D_0. Následně, na základě spočítaného Ljapunova exponentu, odhadneme vzdálenost D_1 těchto dvou křivek v dalším kroku. A v dané vzdálenosti od dalšího bodu na nalezené blízké křivce najdeme nový bod časové řady ve fázovém prostoru. Zpětným převodem souřadnice z fázového prostoru získáme novou hodnotu $x(t+1)$. Postup lze zpřesňovat porovnáním výsledků s využitím jiného, vyššího počtu, dimenzí.

5.5 Další metody predikce

V současném vědeckém zkoumání a publikacích se nejčastěji objevují metody predikce založené na optimalizaci současných a popsaných metod nebo na kombinaci několika dostupných metod vedoucí k zpřesnění predikce. V publikacích je řešena predikce například pomocí skrytých Markovských modelů, které používá k predikci například práce [32]. Příkladem hybridních metod může být například spojení lineární ortogonální kovariance a neuronové sítě [45].

Použitelný pro potřeby predikce pro směrování v překryvných sítích by mohl být i některý z rozšířených Kalmanových filtrů [31], které jsou aplikovatelné i na nelineární systémy.

6 Teze disertační práce

Předmětem aktuálního zkoumání je dynamické směrování v překryvných sítích s využitím predikce. V dosavadních pracích jsem se zaměřoval na dynamické směrování v distribuovaném souborovém systému (DFS). Po původním zavedení směrování v rámci distribuovaného souborového systému[33] jsem navrhl rozšířený algoritmus směrování na základě typu zpráv a jejich klasifikaci do skupin podle požadavků na přenos. Tento algoritmus byl publikován jako teze [34] a následně prakticky ověřen. Realizovaný algoritmus má pozitivní dopad na čas doručení jednotlivých zpráv a také na celkovou odezvou DFS. Výsledky byly publikovány na konferenci[35].

Nejen v distribuovaném souborovém systému lze rozlišovat několik skupin zpráv, jejich požadavků na přenos a dostupných komunikačních kanálů. Rozhodl jsem se proto řešenou oblast zobecnit a aplikovat na širší skupinu problémů. Pro aplikaci dalšího zkoumání jsem zvolil překryvné sítě, kde hledám vhodnou aplikaci známých principů v kombinaci s využitím predikce. Predikce v tomto návrhu umožňuje proaktivně dynamicky měnit směrování, ať už na základě vlastností aktuálních nebo predikovaných parametrů (vlastnosti linek, zatížení distribuované aplikace).

Oblastí, na kterou se aktuálně zaměřuji, je využití různých algoritmů a postupů predikce stavu linek a predikce požadavků na distribuovanou aplikaci při výpočtu směrovacích tabulek. Algoritmy směrování plánuji implementovat a porovnat podle jejich efektivity, využití zdrojů a použitelnosti při efektivním směrování v překryvných sítích. Dále bude zkoumán vliv přesnosti predikce na výsledné směrování (falešné změny ve směrování vlivem lokálních extrémů). Změna směru datového toku není zcela triviální a přináší nutnou režii pro vytvoření a přepnutí přenosových tras, cílem tedy bude i nalezení takové kombinace algoritmů a postupů, které budou tuto režii minimalizovat.

Aktuální publikace se převážně věnují přesnosti predikce, nebo efektivnímu směrování na základě predikce v mobilních nebo optických sítích a internetu[29][23][27][42][3][48][28]. Mobilní sítě jsou překryvným sítím podobné, často zde vznikají a zanikají spojení, uzly se v síti volně stěhují, dochází k častým změnám topologie odchodem nebo příchodem uzlů atp. Optické sítě jsou oproti tomu relativně stabilní a predikce je využita především při přepínání a alokaci přenosových kanálů (Automatically Switched Optical Networks a Generalized Multiprotocol Label Switching).

V disertační práci budou navrženy takové algoritmy, které budou efektivně využívat dostupné výpočetní prostředky s ohledem na přesnost predikce a následné směrování na jejím základě. Navržené algoritmy budou ověřeny na vybrané distribuované aplikaci. Předpokládanou aplikací je distribuovaný souborový systém, který byl vyvíjen v předchozích pracích.

7 Závěr

V odborné práci ke státní doktorské zkoušce byly popsány oblasti, kde se aktuálně překryvné sítě nejvíce uplatňují. Těmito oblastmi jsou především P2P sítě, sítě poskytující komunikační infrastrukturu sítě pro zajištění anonymní komunikace (TOR), sítě poskytující efektivní doručování dat (CDN, CAN), sítě poskytující řízení kvality přenosu (VoIP, IPTV), sítě pro podporu skupinového vysílání (MBone) nebo prostředky pro implementaci nových protokolů a technologií (6BONE).

Následně byly popsány motivace a metody směrování použité v různých druzích překryvných sítí. Každá síť má pro směrování různou motivaci. Peer to peer sítě využívají směrování především k efektivnímu nalezení zdroje v distribuovaném prostředí s co nejnižší složitostí vyhledávání, počtu skoků v prostoru. Anonymizační sítě sledují směrovacími strategiemi především zaručení anonymity a zabezpečení přenášeného obsahu. Sítě pro doručování obsahu nebo nabízející služby garance kvality služeb, kde se především jedná o splnění některého z výkonnostních parametrů, sledují směrovacími strategiemi především splnění těchto požadavků. Vyhledávají a budují spojení, která zajistí co nejrychlejší doručení dat do cíle nebo naopak i přes nižší přenosovou kapacitu vyhledávají spoje podle co nejnižší zpoždění. Dalším sledovaným parametrem, který je pro směrování klíčový, je například stabilita linky a její spolehlivost nebo předpokládaná zátěž překryvné sítě.

Efektivní a pružná reakce je v dalších kapitolách práce podpořena predikcí vlastností jednotlivých linek a možností proaktivně měnit směrovací tabulky a nastavení linek, aby byly splněny požadavky na danou překryvnou síť. V kapitole, která popisovala aktuální metody predikce síťového provozu, byly popsány nejčastěji používané algoritmy, které se v různých vědních oborech používají pro predikci soběpodobných systémů.

Na tuto kapitolu bylo navázáno směrem dalšího výzkumu a cíli disertační práce, kde popisuji další možnosti vývoje a optimalizace existujících postupů. Zvýšení efektivity v případě využití predikce pro směrování v překryvných sítích a zajištění odpovídající kvality služeb hodlám zkoumat především s ohledem na využití výpočetních zdrojů a vliv náročnosti úlohy a vlivu její přesnosti na výsledné směrování.

Reference

[1] David Andersen, Hari Balakrishnan, Frans Kaashoek, and Robert Morris. Resilient overlay networks. *SIGOPS Oper. Syst. Rev.*, 35(5):131–145, October 2001.

[2] Cesnet.cz. Cesnet — prostředí pro vývoj a testování (planetlab), 2015.

[3] J. K. Chiang and Y. H. Lin. A simulation and prediction model for internet traffic and qos based on 1-step markov-chain. In *Computer Modelling and Simulation (UKSim), 2014 UKSim-AMSS 16th International Conference on*, pages 468–473, March 2014.

[4] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Designing Privacy Enhancing Technologies*, pages 46–66. Springer, 2001.

[5] Vlastimil Člupek. *Nelineární analýza a predikce sít'ového provozu*. PhD thesis, Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií, 2012.

[6] M.E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: evidence and possible causes. *Networking, IEEE/ACM Transactions on*, 5(6):835–846, Dec 1997.

[7] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, DTIC Document, 2004.

[8] Hans Eriksson. Mbone: The multicast backbone. *Commun. ACM*, 37(8):54–60, August 1994.

[9] Ashok Erramilli, Matthew Roughan, Darryl Veitch, and Walter Willinger. Self-similar traffic and network dynamics. *Proceedings of the IEEE*, 90(5):800–819, 2002.

[10] Michael J Freedman. Experiences with coralcdn: A five-year operational view. In *NSDI*, pages 95–110, 2010.

[11] Yang Han, K. Koyanagi, T. Tsuchiya, T. Miyosawa, and H. Hirose. A trust-based routing strategy in structured p2p overlay networks. In *Information Networking (ICOIN), 2013 International Conference on*, pages 77–82, Jan 2013.

[12] YT Hou, Z Duan, and Z Zhang. Service overlay networks: Sla, qos and bandwidth provisioning. In *Proc. IEEE ICNP*, volume 2, 2002.

[13] Ma Hui and Yange Chen. Research on p2p routing model based on clustering and position of nodes. In *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, volume 2, pages 307–311, June 2011.

[14] Wang Junsong, Wang Jiukun, Zeng Maohua, and Wang Junjie. Prediction of internet traffic based on elman neural network. In *Control and Decision Conference, 2009. CCDC '09. Chinese*, pages 1248–1252, June 2009.

[15] Jan KACÁLEK and Ivan MÍČA. Nelineární analýza a predikce sít'ového provozu. *VUT v Brně, Elektrorevue*, 2009.

[16] Ryoichi Kawahara, Shigeaki Harada, Noriaki Kamiyama, Tatsuya Mori, Haruhisa Hasegawa, and Akihiro Nakao. Traffic engineering using overlay network. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

[17] Keith Kirkpatrick. Software-defined networking. *Commun. ACM*, 56(9):16–19, September 2013.

[18] John Kubiatowicz, David Bindel, Yan Chen, Steven Czerwinski, Patrick Eaton, Dennis Geels, Ramakrishan Gummadi, Sean Rhea, Hakim Weatherspoon, Westley Weimer, et al. Oceanstore: An architecture for global-scale persistent storage. *ACM Sigplan Notices*, 35(11):190–201, 2000.

[19] Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Trans. Netw.*, 2(1):1–15, February 1994.

[20] Dou Li, Binghui Ji, and Haige Xiang. The on-line prediction of self-similar traffic based on chaos theory. In *2006 International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, 2006.

[21] Michael R Macedonia and Donald P Brutzman. Mbone provides audio and video across the internet. *Computer*, 27(4):30–36, 1994.

[22] Yun Mao, Björn Knutsson, Honghui Lu, and Jonathan M Smith. Dharma: Distributed home agent for robust mobile access. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 2, pages 1196–1206. IEEE, 2005.

[23] P. Millan, C. Molina, E. Medina, D. Vega, R. Meseguer, B. Braem, and C. Blondia. Tracking and predicting link quality in wireless community networks. In *2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 239–244, Oct 2014.

[24] Larry Peterson, Andy Bavier, Marc E. Fiuczynski, and Steve Muir. Experiences building planetlab. In *Proceedings of the 7th Symposium on Operating Systems Design and Implementation*, OSDI '06, pages 351–366, Berkeley, CA, USA, 2006. USENIX Association.

[25] Larry Peterson and Timothy Roscoe. The design principles of planetlab. *ACM SIGOPS operating systems review*, 40(1):11–16, 2006.

[26] Vitaly Petroff. Self-similar network traffic: From chaos and fractals to forecasting and.

[27] L. Pradittasnee. Predicting path quality with cross-layer information in multi-hop wireless networks. In *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pages 464–469, Oct 2015.

[28] B. Puype, E. Marín-Tordera, D. Colle, S. Sánchez-López, M. Pickavet, X. Masip-Bruin, and P. Demeester. Prediction-based routing as rwa in multilayer traffic engineering. *Photonic Network Communications*, 23(2):172–182, 2012. cited By 0.

[29] W. Ramirez, X. Masip-Bruin, E. Marín-Tordera, M. Yannuzzi, A. Martinez, S. Sánchez-López, and V. López. An hybrid prediction-based routing approach for reducing routing inaccuracy in optical transport networks. In *Networks and Optical Communications - (NOC), 2014 19th European Conference on*, pages 147–152, June 2014.

[30] Robert Gentleman Ross Ihaka. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314, 1996.

[31] Dominik Sierociuk and Andrzej Dzielinski. Fractional kalman filter algorithm for the states, parameters and order of fractional system estimation. *International Journal of Applied Mathematics and Computer Science*, 16(1):129, 2006.

[32] R Sivakumar, E Ashok Kumar, and G Sivaradje. Prediction of traffic load in wireless network using time series model. In *Process Automation, Control and Computing (PACC), 2011 International Conference on*, pages 1–6. IEEE, 2011.

[33] Jindřich Skupa. Kivfs - synchronizace a trasování požadavků. Master's thesis, Západočeská univerzita v Plzni, Západočeská univerzita v Plzni. Fakulta aplikovaných věd. Katedra informatiky a výpočetní techniky, 2012.

[34] Jindřich Skupa. Optimalizace synchronizační komunikace v dfs. In *PAD Sborník příspěvků PAD 2014*, pages 117–122. TU Liberec, 2014.

[35] Jindřich Skupa, Luboš Matějka, and Jiří Šafařík. Dynamic internal message routing in distributed file system. In *Scientific Conference on Informatics, 2015 IEEE 13th International*, pages 236–240, Nov 2015.

[36] Neil Spring, Larry Peterson, Andy Bavier, and Vivek Pai. Using planetlab for network research: myths, realities, and best practices. *ACM SIGOPS Operating Systems Review*, 40(1):17–24, 2006.

[37] P. Syverson. Onion routing for resistance to traffic analysis. In *DARPA Information Survivability Conference and Exposition, 2003. Proceedings*, volume 2, pages 108–110 vol.2, April 2003.

[38] G. Terdik and T. Gyires. Does the internet still demonstrate fractal nature? In *Networks, 2009. ICN '09. Eighth International Conference on*, pages 30–34, March 2009.

[39] Jelle van den Hooff, David Lazar, Matei Zaharia, and Nickolai Zeldovich. Vuvuzela: Scalable private messaging resistant to traffic analysis. In *Proceedings of the 25th Symposium on Operating Systems Principles*, SOSP '15, pages 137–152, New York, NY, USA, 2015. ACM.

[40] Andras Veres and Miklos Boda. The chaotic nature of tcp congestion control. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1715–1723. IEEE, 2000.

[41] Xuqi Wang, Chuanlei Zhang, and Shanwen Zhang. Modified elman neural network and its application to network traffic prediction. In *Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on*, volume 02, pages 629–633, Oct 2012.

[42] Y. Wei and J. Wang. A delay/disruption tolerant routing algorithm based on traffic prediction. In *The 27th Chinese Control and Decision Conference (2015 CCDC)*, pages 3253–3258, May 2015.

[43] W. Willinger. Self-similarity in wide-area network traffic. In *Lasers and Electro-Optics Society Annual Meeting, 1997. LEOS '97 10th Annual Meeting. Conference Proceedings., IEEE*, volume 2, pages 462–463 vol.2, Nov 1997.

[44] Shun-an Wu, Qiao Yan, Xue-song Qiu, and Yanjie Ren. A probe prediction approach to overlay network monitoring. In *Proceedings of the 7th International Conference on Network and Services Management*, pages 465–469. International Federation for Information Processing, 2011.

[45] Lin Xiang, Xiao-Hu Ge, Chuang Liu, Lei Shu, and Cheng-Xiang Wang. A new hybrid network traffic prediction method. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–5. IEEE, 2010.

[46] Xipeng Xiao, A. Hannan, B. Bailey, and L.M. Ni. Traffic engineering with mpls in the internet. *Network, IEEE*, 14(2):28–33, Mar 2000.

[47] Zhiyong Xu, Rui Min, and Yiming Hu. Hieras: a dht based hierarchical p2p routing algorithm. In *Parallel Processing, 2003. Proceedings. 2003 International Conference on*, pages 187–194, Oct 2003.

[48] Zhi yuan LI, Ru chuan WANG, Zhi jie HAN, Jun lei BI, and Chong HAN. Traffic prediction-based routing algorithm over structured p2p networks. *The Journal of China Universities of Posts and Telecommunications*, 18, Supplement 1:23 – 27, 2011.

[49] Jun Zhang, KC Lam, WJ Yan, Hang Gao, and Yuan Li. Time series prediction using lyapunov exponents in embedding phase space. *Computers & Electrical Engineering*, 30(1):1–15, 2004.