Viivesietoisten verkkojen nykytila ja tulevaisuuden haasteet

Seppo Syrjänen

Helsinki 23.11.2007

Pro gradu -tutkielma

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta/Osasto — Fakultet/Sektion — Faculty	Laitos — Institution — Department
Matemaattis-luonnontieteellinen tiedekunta	Tietojenkäsittelytieteen laitos

Tekijä — Författare — Author
Seppo Syrjänen

Työn nimi — Arbetets titel — Title

Viivesietoisten verkkojen nykytila ja tulevaisuuden haasteet

Oppiaine — Läroämne — Subject
Tietojenkäsittelytiede

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Pro gradu -tutkielma	23.11.2007	72 sivua + 6 liitesivua

Tiivistelmä — Referat — Abstract

Viive- ja häiriösietoiset verkot (Delay/Disruption Tolerant Networks, DTN) ovat tietoliikenneverkkoja, joissa siirtoviiveet ja virhetiheydet ovat suuria sekä yhteyskatkot tavallisia. DTN-verkoissa toimivat sovellukset ovat mukautuneet yhteyskatkoihin sekä päästä päähän -yhteyden puuttumiseen. Näille sovelluksille riittää, että viesti toimitetaan perille esimerkiksi tiettyyn aikaan mennessä.

DTN-verkkojen virheiden sietokyky perustuu viestien siirtämiseen verkossa yksi solmuväli kerrallaan ja viestien tallettamiseen verkkosolmuissa haihtumattomalle tallennusvälineelle. Näin verkon viestinvälitys voi toimia pitkistäkin yhteyskatkoista ja solmujen uudelleenkäynnistyksistä huolimatta.

Viivesietoisten verkkojen sovelluksia ovat esimerkiksi yhteydet avaruusluotaimiin toisilla planeetoilla tai viestinvälitys seuduilla, joilla ei ole tarjolla kiinteää tietoliikenneinfrastruktuuria. Muita sovellusalueita ovat pelastus- ja sotilasyhteydet, sensoriverkot sekä liikkuvien käyttäjien ja ajoneuvojen verkot.

Tässä tutkielmassa esitellään viivesietoisten verkkojen arkkitehtuurin perusratkaisuja sekä joitakin sovellusalueita. Erityisesti käsitellään reititystä sekä sen resurssinkulutuksen pienentämiseen kehitettyjä ratkaisuja. Lisäksi tutkielmassa esitellään viivesietoisten verkkojen ja niiden sovellusalueiden tulevaisuudennäkymiä sekä alan uusimpien tutkimustulosten että aktiivitutkijan haastattelun avulla.

ACM Computing Classification System (CCS):
C.2.1 [Network Architecture and Design]

Avainsanat — Nyckelord — Keywords
Viivesietoiset verkot, Häiriösietoiset verkot, Delay/Disruption Tolerant Networks, DTN

Säilytyspaikka — Förvaringsställe — Where deposited
Kumpulan tiedekirjasto, sarjanumero C-

Muita tietoja — övriga uppgifter — Additional information

Sisältö

Alkusanat

Kiinnostuin viivesietoisista verkoista ensimmäistä kertaa vuonna 2001 ollessani työmatkalla konferenssissa, jossa Vint Cerf hahmotteli planeettainvälisen Internetin toteutusta. Entisenä tähtitietelijänä kuuntelin esitelmää ilahtuneena siitä, että jotkut katsovat verkkoasioissakin todella kauas.

Muutama vuosi myöhemmin löysin itseni jatkamassa keskenjääneitä opintojani huomattavan pitkän tauon jälkeen. Opintojen loppuvaiheessa osallistuin professori Jukka Mannerin New Internet Technologies -seminaariin, jossa eräänä aineena olivat DTN-verkot. Planeettainvälinen Internet oli kuluneina vuosina kasvanut korkoa ja saanut seurakseen kaksi muuta hienolta tuntuvaa sovellusalaa: sensoriverkot maapallon tilan mittaamiseen sekä kehittyvien maiden tietoliikenteen parantamisen. Tartuin näihinkin verkkoihin niin tiukasti, että jatkoin seminaarin jälkeen pro gradu -tutkielmaa samasta aiheesta.

Tämänkin opinnäytteen valmistuminen vaati tukea lukemattomilta tahoilta, joista haluan mainita omenanviljelijät, kahvinmaustajat, viskinsavustajat, energiantölkittäjät, Nutriksen sekä Yliopistoliikunnan. Henkilökohtaisemmalla tasolla kiitän ohjaajaani Jukka Manneria rauhoittavista neuvoista, DTN-guru Jörg Ottia luvun 4 haastattelusta sekä opiskelukavereitani Saria, Sannaa ja Mikkoa verrattomasta vertaistuesta. Lisäksi lausun todella lämpimät kiitokseni työpaikalleni, ystävilleni ja perheilleni tarpeellisista piipahduksista todellisuuteen sekä vaimokullalleni aivan kaikesta.

> *Kaiken tässä tutkielmassa olevan olen lukenut jostakin. Paitsi ne kohdat, jotka keksin itse.* –Iain Banksiä mukaillen

1 Johdanto

Maailmanlaajuisen Internetin menestys perustuu laajassa käytössä oleville avoimille protokollille. Internetin verkkopakettien välittämiseen perustuva toimintatapa on kevyt ja yksinkertainen toteuttaa hyvin erilaisiin laite- ja verkkoympäristöihin. Nykyisin Internetin protokollat ovatkin käytössä kaikilla elämänaloilla hyvin monenlaisten verkkopohjaisten sovellusten alustana.

Internet-protokollat toimivat kuitenkin vain verkoissa, joiden siirtonopeudet, virhetiheydet ja kiertoviiveet vastaavat niitä tietoliikennetekniikoita, joiden ehdoilla protokollia on kehitetty. Suuret poikkeamat näissä parametreissa voivat estää protokollien toiminnan kokonaan, tai ainakin merkittävästi heikentää niiden tehokkuutta.

Internetin verkkotason protokolla IP [Pos81] perustuu verkkopakettien välittämiseen lähettäjältä vastaanottajalle. IP-protokolla on sinänsä yhteydetön, mutta pakettien on päästävä perille vastaanottajalle siirtoviiveen sallimalla tahdilla, jota lähettäjän ja vastaanottajan välille muodostuu yhteys. Jos tällaista päästä päähän -yhteyttä ei ole olemassa, eli jos lähettäjän paketit eivät pääse vastaanottajalle, ei IP-protokollakaan voi toimia.

Toinen Internet-protokollien olettamus verkkoyhteyden laadulle on tarpeeksi pieni toimitusaika. Paketin tulee päästä vastaanottajalle käytännössä reaaliajassa. Kahdensuuntaisessa viestinnässä myöskään viestin kuittauksen paluuseen lähettäjälle ei saa kulua merkittävästi aikaa. Kiertoviive (engl. round trip time, RTT) on aika, jona viesti ja sen kuittaus kulkevat verkossa kumpaankin suuntaan. Jos kiertoviive on liian suuri, eivät protokollat tai niiden päällä olevat sovellukset voi voimia. Yksittäisen sanoman tai kuittauksen satunnainen katoaminen matkalla ei sinänsä ole ongelma, sillä protokollat ja sovellukset toipuvat siitä uudelleenlähetyksellä [APS99]. Internetin yhteyskerros TCP kuitenkin tulkitsee pakettien katoamisen merkiksi verkon ruuhkaantumisesta ja pienentää lähetysnopeutta. Tämä on haitallista, jos paketteja on kadonnut verkosta vain siirtohäiriöiden vuoksi [RKMF03].

Käytännössä myös kaistanleveyden kumpaankin suuntaan tulee olla suunnilleen samansuuruinen. Lähettäjä ja vastaanottaja voivat olettaa voivansa lähettää yhtä suuria määriä liikennettä yhteyden toiseen päähän. Merkittävä epäsymmetria yhteyden kapasiteetissa tai lähetysvuorojen jaossa johtaa verkon ruuhkaantumiseen [FLZ+05, XS02, PWWM05].

Edellä mainittujen olettamuksien lisäksi Internet-protokollat lakkaavat toimimasta tehokkaasti, jos yhteys tuottaa paljon siirtovirheitä tai muuttaa merkittävästi IP-pakettien järjestystä. Internetin protokollat olettavat myös, että sovelluksien ei tarvitse välittää alla olevan verkon toiminnasta, että riittävä turvallisuus ja käytettävyys saadaan aikaan päästä pää-

hän toimivilla mekanismeilla ja että riittävä tehokkuus saavutetaan löytämällä yksi reitti lähettäjältä vastaanottajalle.

Etenkin langattomien tietoliikennetekniikoiden yleistyminen on tuonut entistä laajempaan käyttöön verkkoja ja sovelluksia, jotka eivät joko toimintatavaltaan tai toimintaympäristöltään sovellu Internetin protokollille. Näissä ns. haasteellisissa verkoissa (engl. challenged networks) on sovellusalueita, joissa niiden käyttö on jopa mahdotonta.

Eräs tällainen sovellusalue ovat hätäyhteyksiin tai sotilaskäyttöön tarkoitetut viestintäverkot. Niiden halutaan mahdollistavan viestinvälitys esimerkiksi pelastustehtävissä tai taistelutilanteissa, joissa ei ole käytettävissä mitään viestintäinfrastruktuuria [MED07]. Toinen samantapainen käyttötapa ovat teknologisesti matalalla tasolla olevien yhteisöjen verkkoyhteydet esimerkiksi Aasian maaseutukylissä. Viranomaisten peruspalveluiden ja tiedonvälityksen toteuttaminen satelliittiyhteyksillä tai epäluotettavilla puhelinlinjoilla on epäkäytännöllistä ja usein myös kallista.

Hyvin monenlaisia geo- ja biotieteiden tutkimusongelmia tutkitaan sijoittamalla tutkittavalle alueelle lukuisia mittalaitteita eli sensoreita rekisteröimään haluttujen suureiden muuttumista ajan ja paikan funktiona [LGE06, MCP+02, HM+06, WW07]. Näiden sensoriverkkojen (engl. sensor networks) solmujen keräämät tiedot joudutaan yleensä siirtämään keräysasemalle kuljettamalla ne askel askeleelta verkon solmulta toiselle. Solmujen niukat energia- ja talletusresurssit sekä epäluotettavat yhteydet vaikeuttavat Internetprotokollien käyttöä, sillä päästä päähän -yhteyden muodostaminen useasta epäluotettavasta yhteysvälistä on hyvin epävarmaa [LGE06].

Samanlaisia suoria laitteiden välisiä langattomia yhteyksiä voitaisiin käyttää myös täydentämään infrastruktuuriverkon palveluita esimerkiksi siten, että kannettavat laitteet välittäisivät toistensa liikennettä alueilla, jossa langattoman palveluverkon tukiasemia on harvassa. Näin osa sovelluksista säilyisi käyttökelpoisena, vaikka infrastruktuurin tarjoamat yhteydet eivät olisikaan täysin kattavia. Tästä hyötyisivät esimerkiksi nomadiset käyttäjäryhmät ja ajoneuvopohjaisten verkkojen sovellukset [BGJL06, OK04, HKL+07].

Osalle sovelluksista riittää Internetin palvelumallia kevyempikin tapa saada viesti perille. Esimerkiksi tekstipohjaiseen pikaviestintään mobiililaitteiden välillä tai sähköpostien siirtoon ei tarvita reaaliaikaisuuteen pystyvää jatkuvasti käytössä olevaa yhteyttä Internetiin tai palveluntarjoajan verkkoon. Vaikka viestit välitetään verkkoprotokollien aikaskaalassa hitaasti, voi esimerkiksi muutamien sekuntien toimitusviive olla inhimillisen käyttäjän kannalta täysin riittävä. Tällaisen viestintäyhteyden toteuttamiseen riittää huomattavasti perinteistä kevyempi infrastruktuuri.

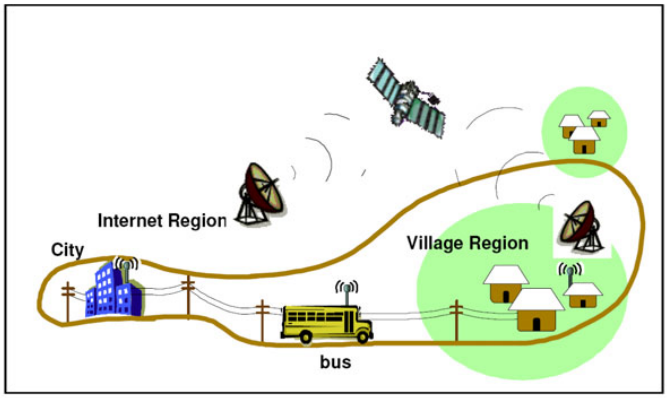
Kaikkein suurimmat toimintaympäristöstä johtuvat haasteensa Internet-protokollat kohtaavat avaruustutkimuksessa. Internet-yhteydet toisille planeetoille eivät yksinkertaisesti ole mahdollisia tähtitieteellisten etäisyyksien ja radiosignaalien äärellisen nopeuden vuoksi [IPN]. Internetin siirtokerroksen protokolla TCP [CDS74] on tehty maanpäällisiin verkko-olosuhteisiin ja toimii vain jos kiertoviive on tarpeeksi pieni [FCG06, WPP+07]

Vaikka edellä esitellyt haasteellisten verkkojen sovellusalueet ovatkin kaukana toisistaan, voidaan niiden kaikkien ominaisuuksia ja vaatimuksia tietoliikenteelle mallintaa likimain samoilla tavoilla. Tästä oivalluksesta on syntynyt viive- ja häiriösietoisten verkkojen (engl. Delay/Disruption Tolerant Networks, DTN) tutkimusala sekä yhteisten standardien kehitys [CBD+07, Fall03, SB07].

DTN-verkoissa kohtaavat liikkuvien tilapäis- eli ad hoc -verkkojen (engl. mobile ad-hoc networks, MANET), liikkuvien verkkojen (engl. network mobility, NEMO), sensoriverkkojen, langattomien verkkojen sekä uuden tietoliikennetekniikan tutkimus ja soveltaminen. DTN-verkkojen käyttötapaukset ovat vielä totuttua haastavampia, sillä niissä varaudutaan jatkuviin yhteyskatkoihin, suuriin virhemääriin, niukkoihin energia- ja talletusresursseihin sekä pitkiin kiertoviiveisiin.

DTN-verkon palvelumalli on väljempi, laajempi ja yleiskäyttöisempi kuin Internetin IP-paketteja päästä päähän välittävä toimintatapa. Ehdotetussa DTN-verkkojen arkkitehtuurissa perinteisten verkkojen päälle muodostetaan sovellustason viestinvälitysrajapinta, jossa välitetään sovellusten kannalta mielekkäitä tietokokonaisuuksia, viestinippuja (engl. bundle). Viestit siirretään verkossa mahdollisesti vain yksi solmuväli kerrallaan ja talletetaan välittäviin solmuihin (engl. store-and-forward). Verkon jokainen solmu on myös DTN-reititin, perinteisten ad hoc -verkkojen tapaan. Solmujen välisten yhteyksien opportunistinen käyttö yhdistettynä toimitusvastuun siirtämiseen (engl. custody transfer) sekä liikkuvien solmujen mukanaan kuljettamiin välitettäviin viesteihin (engl. store-carry-and-forward) luovat pohjan DTN-arkkitehtuurin viive- ja vikasietoisuudelle.

Verkkosolmujen resurssinkulutus viestien tallettamisessa ja välittämisessä vaikuttaa koko verkon suorituskykyyn. Solmut voivat kertoa viestiensä kelvollisen elinajan ja kiireellisyysluokan, joiden mukaan välittävät solmut voivat valita verkosta pudotettavat viestit paremmin.



Kuva 1: Esimerkki viivesietoisten verkkojen sovelluksesta. Syrjäseutujen sähköposti- ja WWW-kioskien liittäminen Internetiin erilaisten yhteystapojen avulla [DBF04].

DTN-reitityksen protokollat pystyvät hyödyntämään kaikki saamansa tilaisuudet lähettää viestejä eteenpäin sekä valitsemaan useiden suorituskyvyltään tai kustannuksiltaan erilaisten yhteystapojen väliltä. Kuvassa 1 on esimerkki kylästä, jonka viestiliikennettä voidaan välittää puhelinlinjojen ja satelliittiyhteyksien lisäksi myös viestinvälityssolmun sisältävällä reittibussilla. Viesteistä voidaan lähettää verkkoon myös useita kopioita, mahdollisesti erilaisten yhteystapojen kautta, jos halutaan varmistaa edes yhden niistä menevän perille.

Oma tutkimusalansa ovat myös viivesietoisten verkkojen sovellukset. Tähän asti jokainen haasteellisten verkkojen sovellusalue on kehittänyt omat suljetut protokollansa ja sovelluksensa. Yhteisillä standardeilla voidaan helpottaa sekä näiden sovelluksien tuottamista että mahdollistaa niiden yhteiskäyttö [Goth06]. DTN-verkkojen viestinvälitysrajapintaa käyttäville sovelluksille tarjotaan mahdollisuus ottaa kantaa verkon toimintaan ja resurssikulutukseen. Perinteisessä verkkosovellusarkkitehtuurissa sovelluskerroksella ei ole sovellusrajapintaan kuuluvaa mahdollisuutta olla tietoinen tai vaikuttaa alla olevan verkkoyhteyden toimintaan.

DTN-verkkojen voidaan ajatella pyrkivän kattamaan olemassa olevien verkkotekniikoiden jättämät aukot. Tutkimuksen tavoitteena voidaan nähdä tietoliikenteen palvelumalli, joka pystyy tarjoamaan viestin välittämisen lähettäjältä vastaanottajalle hyvin erilaisilla toimitustavoilla ja siirtotekniikoilla. DTN-tekniikalla voidaan toteuttaa uudenlaisia sovelluksia ympäristöihin, joissa perinteiset verkkosovellukset eivät toimi.

DTN-arkkitehtuurityön tuloksena on uusi OSI-mallia ratkaisevasti täydentävä yleiskäyttöinen verkkosovellusten arkkitehtuuri, jonka käyttämisestä hyvillä verkkoyhteyksillä ei ole mitään haittaa, mutta joka huonoissa olosuhteissa toimii ratkaisevasti nykyisiä tekniі-

koita paremmin. Tähän tavoitteeseen pääsemiseksi tarvitaan vielä runsaasti tutkimusta ja käytännön pilottisovelluksia, sillä DTN-verkoissa yhdistyvät usean haastavan sovellusalueen vaikeimmat ongelmat.

Tässä tutkielmassa esitellään viive- ja häiriösietoisten verkkojen tutkimustuloksia, protokollia ja sovelluksia. Luvussa 2 esitellään DTN-verkkojen arkkitehtuuri ja toiminta nykyisen arkkitehtuurimallin mukaan. Luvussa 3 perehdytään laajalti DTN-verkkojen tutkimuksen suurimpaan haasteeseen eli reititykseen ja viestinvälitykseen verkon solmujen välillä. Luvussa 4 pohditaan DTN-verkkojen tulevaisuutta sekä uusimpien tutkimustulosten että alan aktiivitutkijan haastattelun kautta. Lopuksi luvussa 5 tarjotaan yhteenveto.

2 DTN-verkkojen arkkitehtuuri

Viivesietoisten verkkojen tutkimusta koordinoi Internet Research Task Force -organisaation DTN Research Group -ryhmä [IRTF, DTNRG]. Vastaavaa tutkimusta tekevät myös Yhdysvaltain ilmailu- ja avaruushallinto NASA sekä useiden maiden puolustusministeriöt [BGJL06, RHB+07]. Tutkimuksen tavoitteena on selkeä sovellusarkkitehtuuri, joka parantaa liikennöintiä silloin, kun yhteydet ovat ajoittaisia ja alttiita häiriöille tai kun viestin perillemenoon tarvitaan useita heterogeenisia, Internetin protokollille soveltumattomia verkkoratkaisuja [Fall03, War03, FD06, BEF+05].

Keväällä 2007 julkaistuun DTN-verkkojen arkkitehtuuria kuvaavaan dokumenttiin RFC4838 [CBD+07] on yhdistetty eri sovellusalueisiin kuuluvien haasteellisten verkkojen tarpeita. DTN-verkon palvelumalli on laajempi ja yleiskäyttöisempi, kuin Internetin IP-paketteja päästä päähän -palvelumalli. Se myös keventää perinteisten verkkosovellusten edellyttämää jatkuvan päästä päähän -yhteyden vaatimusta verkkoyhteydelle. Tässä luvussa esitellään ehdotetun DTN-arkkitehtuurin peruspiirteet [CBD+07, Fall03, War03, FD06, SB07].

2.1 Sovellustason viestinvaihtoprotokolla

Toisin kuin Internetissä, jossa välitetään paketteja ja bittivirtaa jatkuvan verkkoyhteyden päällä, välitetään DTN-verkoissa sovellusten kannalta mielekkäitä, mahdollisesti hyvinkin suuria tietokokonaisuuksia, viestinippuja (engl. bundle). DTN-verkko on sovellustason kateverkko (engl. overlay network), joka nimensä mukaisesti kattaa erilaisten tiedonsiirtotapojen ja -protokollien toteutuksen yksityiskohdat. DTN-verkot voivat rakentua perinteisen elektronisen tai optisen tiedonvälityksen lisäksi myös viestien siirtämiselle

fyysisillä tallennusvälineillä [SKZ06].

Viestit siirretään verkossa mahdollisesti vain yksi solmuväli kerrallaan (engl. hop-by-hop). Vastaanotetut viestit talletetaan välittäviin solmuihin odottamaan tulevia yhteysmahdollisuuksia (engl. store-and-forward). Verkon jokainen solmu on DTN-reititin, mutta eri solmut voivat toimia reitityksessä erilaisilla rooleilla välityskapasiteettinsa mukaan. Viestit välitetään DTN-reitittimeltä toiselle, kunnes ne saavuttavat vastaanottajan. Välissä voi olla useita erillisiä, tekniikaltaan erilaisia verkkoja.

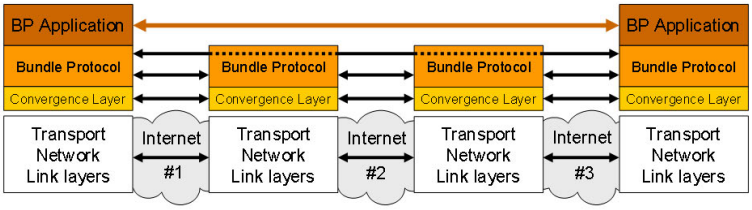
Tämä solmujen tekemä tallennus tapahtuu aikaskaalaltaan eri alueella, kuin tavallisten verkkojen muutamia millisekunteja kestävä verkkopaketin talletus muistiin reitityspäätöksen ajaksi. Viesti saattaa olla solmun pysyväismuistissa odottamassa edelleenlähetystä mielivaltaisen pitkän ajan, ellei viestin voimassaoloaika muuta määrittele.

DTN-solmut voivat myös itse liikkua ja kuljettaa viestejä mukanaan. Siinä missä liikkuvuus on muissa verkoissa ongelma, edellyttää osa DTN-verkkojen käyttötapauksista solmujen liikkuvan ja kuljettavan viestejä mukanaan (engl. store-carry-and-forward).

Erikseen viestien kuljettamiseen tarkoitettua solmua kutsutaan viestilossiksi tai tietojuhdaksi (engl. message ferry, data mule) [PFH04, TAZ06]. Näin esimerkiksi kylän halki ajava linja-auto tai konferenssiosallistujan älypuhelin voivat kuljettaa mukanaan ja välittää edelleen toisten keskusteluosapuolien viestejä [DFL01, HCS+05]. Sensoriverkon solmujen keräämiä tietoja voidaan purkaa tutkimusalueen läpi liikkuvalla keräyssolmulla, jolloin voidaan hyödyntää myös lyhyemmän lähetysmatkan tuomia etuja eli lähettää suurempia tietomääriä pienemmällä lähetysteholla.

Viestin toimitusvastuu voidaan siirtää verkossa eteenpäin (engl. custody transfer) esimerkiksi kaikkein virhealtteimman yhteysvälin päissä oleville solmuille. Tällöin uudelleenlähetyksiä ei turhaan tehdä päästä päähän lähettäjän ja vastaanottajan välillä, vaan siinä kohdassa verkkoa, jossa ongelmia eniten esiintyy.

Sama viesti voidaan lähettää verkkoon useita kertoja ja mahdollisesti myös rinnakkaisia polkuja pitkin, etenkin jos sen oletetaan parantavan viestin mahdollisuuksia päästä perille. Useita eri yhteystapoja voidaan käyttää rinnakkain myös niin, että suurin osa liikenteestä toimitetaan perille fyysisellä medialla ja virhealttiita langattomia yhteyksiä käytetään vain hallintaliikenteeseen [HA06]. Internetissä kukin paketti lähetetään verkkoon vain kerran, sillä vaikka uudelleenlähetyksissä lähetetään kyllä sovelluksen kannalta sama tieto, itse paketti ei ole sama. Samoin IP-paketit kulkevat lähettäjältä vastaanottajalle käytännössä vain yhtä reittiä pitkin.



Kuva 2: Viestien välitys DTN-arkkitehtuurin Bundle-protokollassa tapahtuu sovellusten välillä. Viesti voi kulkea erilaisten verkkojen välillä niitä yhdistävien DTN-reitittimien kautta [NETLAB].

Kuva 2 esittää kahden DTN-sovelluksen välistä vuorovaikutusta useista erillisistä verkoista koostuvan yhteyden yli. Eri verkoissa käytetään erilaisia siirtokerroksen protokollia, jotka liitetään omalla sovituskerroksellaan (engl. convergence layer) DTN-arkkitehtuurin viestinvälityskerrokseen. Käytännön toteutuksia sovituskerroksesta on vasta TCP/IP-protokollalle [DO07], mutta luonnoksia on tehty myös satelliitti- ja pelastussovelluksiin.

Tavallinen Internetin nimipalvelu perustuu lähettäjän tekemään nimiselvitykseen vastaanottajan nimestä kohdekoneen verkko-osoitteeksi. Tämän toiminta edellyttää yhteyttä maailmanlaajuiseen Internetin nimipalveluun (engl. Domain Name System, DNS) [Moc87]. Vastaanottajan verkko-osoitteen selvittäminen lähetysvaiheessa (engl. early binding) ei ole mahdollista viivesietoisissa verkoissa, koska lähettävällä sovelluksella ei välttämättä ole edes yhteyttä verkkoon viestin lähetyshetkellä.

DTN-verkkojen sovellusarkkitehtuuri määrittelee geneerisen ja laajennettavan tavan nimetä eri verkkoja, niiden jäseniä sekä sovelluksia. DTN-verkoissa käytetään URI-muotoisia päätetunnisteita (engl. End System Identifier, EID) ja myöhäistä nimiselvitystä (engl. late binding). Näin vastaanottajan verkko-osoite selvitetään mahdollisimman lähellä vastaanottajaa, mahdollisesti vasta edellisessä solmussa [FCG06]. Tällaisen toimintatavan avulla viestejä voidaan lähettää hyvin erilaisiin verkkoihin ja nimentäalueisiin. Myöhästetty nimiselvitys jättää tilaa tulevien verkkojen vielä määrittelemättömille nimentätavoille. Riittää, että käytettävissä on eri verkkoja ja nimiavaruuksia liittäviä välityssolmuja, joiden kautta vanhatkin verkot voivat lähettää viestejä uusiin verkkoihin.

Päätetunnisteet kuvaavat yksittäisten solmujen sijaan abstraktimmin mahdollisesti useammassakin solmussa toimivia sovelluksia, joten viestin tulkitaan päässeen perille, jos se on toimitettu edes yhteen kyseisen tunnisteen vastaanottajaksi rekisteröityneeseen solmuun. EID-tunnisteiden voi siten ajatella toimivan ryhmä- tai jokulähetyksen (engl. multicast, anycast) tapaan, yksittäisistä solmuista riippumatta. Tämä lisää merkittävästi viestinnän luotettavuutta ja vikasietoisuutta.

Taulukko 1: Kaikissa DTN-viesteissä olevat pakolliset otsikkokentät yksilöivät viestin sekä ohjaavat sen käsittelyä verkossa. EID (Endpoing Identifier) kertoo viestin lähettäjä- ja vastaanottajasovelluksen tunnisteen. [SB07]

Otsikkokenttä	**Tarkoitus**
Aikaleima	Yksilöi viestin sen lähettäjäsolmun EID:n kanssa. Aikaleimana käytetään sovelluksen lähetyspyyntöhetken aikaa, ei viestin lähetysaikaa. Aikaleimojen käyttö edellyttää verkolta aikasynkronointia.
Elinikä	Viestin eliniän kuluttua sen saa poistaa verkosta.
Palveluluokka	Mihin palveluluokkaan (bulkki, normaali, pika) viesti kuuluu.
Lähettäjän EID	Viestin lähettäneen sovelluksen tunniste.
Vastaanottajan EID	Viestin vastaanottajan tunniste.
Toimitusraporttien EID	Mille solmulle lähetetään raportit viestin toimittamisesta perille. Tämä voi olla eri kuin lähettäjä.
Uudelleenlähettäjän EID	Viestin uudelleenlähetyksistä tällä hetkellä huolehtiva solmu (engl. custodian).

2.2 Viestien käsittely

DTN-verkossa välitettävien viestien rakenne muistuttaa IPv6-protokollan [DH98] otsikkokenttien ketjuttamista. Jokaisessa viestissä on joukko pakollisia otsikkokenttiä ja niiden lisäksi joukko valinnaisia kenttiä. Kaikissa viesteissä olevat pakolliset kentät on esitelty taulukossa 1.

DTN-verkkojen viestinvälitysprotokolla pyrkii maksimoimaan käytettävän kaistanleveyden ja parantamaan viestin perillemenon todennäköisyyttä määrittelemällä kaksi tapaa pilkkoa viestinippuja pienempiin osiin. Ennakoivassa pilkkomisessa (engl. proactive fragmenting) suuri viesti pilkotaan pienempiin osiin esimerkiksi siksi, että se mahtuisi kerrallaan käytössä olevaan kontaktiin. Jokaisesta viestin osasta tulee oma itsenäinen viestinsä, joka reititetään verkon läpi muista osista riippumatta. Reagoiva pilkkominen (engl. reactive fragmenting) toimii samaan tapaan kuin Selective ACK uudemmissa TCP-toteutuksissa, mutta lähettäjän kannalta katsottuna. Joissakin tilanteissa lähettävä solmu voi tietää, että vain osa lähetetystä viestistä on päässyt perille ehjänä. Tällöin lähettäjä voi muodostaa jäljelle jääneestä osasta uuden viestin lähetettäväksi seuraavan kontaktin aikana. Näin alkuperäisen viestin alkuosaa ei tarvitse lähettää enää uudestaan.

Viestien pilkkominen tapahtuu jokaisen solmun paikallisen arvion pohjalta, eikä siis esimerkiksi siirtotien kehyskoon (engl. medium transfer unit, MTU) tai muun ulkoisen tekijän perusteella. Jos solmun mielestä viestin pilkkominen parantaa sen mahdollisuuksia päästä perille, se on vapaa tekemään niin.

Vastaavalla tavalla välittävä solmu voi vapaasti siirtää viestinippuja eri reititysjonoista toiseen esimerkiksi siksi, että niitä ei ole saatu perille jokin tietyn verkkoliittymän kautta määräaikaan mennessä. Saman alkuperäisen viestin osia saa myös vapaasti yhdistää takaisin isommiksi kokonaisuuksiksi matkan varrella, jos tilaisuus siihen tarjoutuu. Näin välityssolmu pääsee käsittelemästä jokaista viestin osaa erikseen, vaan voi lähettää ne yhtenä viestinä sopivassa kontaktissa.

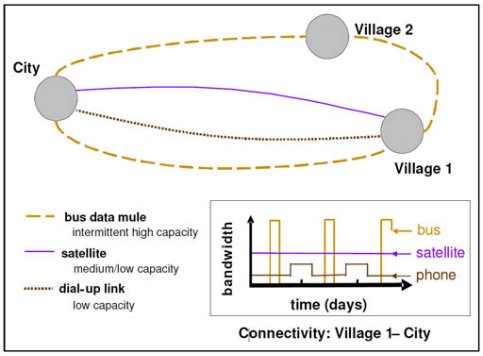
2.3 Reititys

Solmujen välisten yhteyksien ennustamattomuuden, katkojen ja jatkuvan yhteyden puutteen vuoksi viivesietoisten verkkojen reititys poikkeaa perinteisestä kiinteiden verkkojen reitityksestä. Eniten yhtymäkohtia on löydettävissä itseorganisoituvien tilapäisverkkojen eli ad hoc -verkkojen reitityksestä.

Reitityksen perusongelmat ovat DTN-verkoissa sikäli samat, kuin muissakin verkoissa, eli milloin lähettää ja vastaanottaa viestejä, keneltä niitä hyväksyä ja kenelle antaa välitettäväksi. Välitettävien viestien valinta sekä talletettavien viestien poistaminen ruuhkatilanteessa ovat DTN-verkoissa vielä tavallista suurempia ongelmia, sillä koko verkon toiminta perustuu viestien tallettamiseen verkon solmuihin.

DTN-verkkojen reititysmalli jakaa solmujen väliset yhteydet eli kontaktit (engl. contact) pysyviin (engl. persistent), tilattaviin (engl. on-demand) sekä ajoittaisiin (engl. intermittent). Tilattavat kontaktit ovat mahdollisia solmun aktiivisilla toimilla, esimerkiksi soittamalla yhteys auki modeemilla. Ajoittaiset kontaktit eivät ole solmun omassa hallinnassa, vaan saattavat tapahtua ajastetusti jonkin aikataulun mukaan (engl. scheduled), ennustetusti aikaisempien kontaktien perusteella päättelemällä (engl. predicted) tai opportunistisesti eli ilman ennakkovaroitusta (engl. opportunistic).

Ajastettuja kontakteja ovat esimerkiksi yhteydet satelliittien tai tietolossien kautta. Verkon solmut tietävät ennakolta laaditun aikataulun mukaan, milloin yhteys on saatavilla. Ennustettujen kontaktien käyttäminen taas perustuu siihen, että solmut oppivat edellisistä kohtaamisista toistensa liikkumistapoja (engl. mobility model), joiden avulla ne osaavat varautua tulevaan kontaktiin. Satunnaiset, odottamattomat kontaktit ovat solmujen kannalta haastavimpia, sillä solmujen pitää niitä käyttääkseen olla koko ajan valmiina yhtey-



Kuva 3: DTN-reititys ottaa huomioon erilaisten kontaktien kapasiteetin ja hinnan. Puhelinyhteys on edullisempi öisin. Bussi kulkee kylän kautta harvoin, mutta kuljettaa paljon tietoa. Satelliittiyhteys on tarjolla koko ajan, mutta se on kallis [DBF04].

teen, jonka kestostakaan ei ole varmuutta. Satunnaiset kontaktit ovat reitityksen kannalta mielenkiintoisimpia, sillä niiden avulla solmut voivat toimia ilman mitään ennakkotietoa verkon topologiasta tai toistensa liikkeistä.

DTN-verkon reitityksessä voidaan mallintaa erilaisia kontakteja kustannuksineen. Kuvan 3 oikean alareunan kaavio kuvaa eri yhteysvaihtoehtojen ajallista saatavuutta sekä kapasiteettia. Tavallinen viestiliikenne voi hyvin odottaa seuraavaa välittäjäbussia, mutta pienet kiireelliset viestit voidaan lähettää koko ajan saatavilla olevalla satelliittiyhteydellä.

Viestien välitys verkossa (engl. forwarding) on sitä tehokkaampaa, mitä paremman tilannekuvan valittu reititysmalli muodostaa verkon olemassa olevista tai tulevista yhteyksistä. Jos mitään tietoa ei ole käytettävissä, joudutaan turvautumaan välitysmalleista yksinkertaisimpaan, epideemiseen reititykseen. Epideemisessä reitityksessä viesti lähetetään jokaiselle kohdatulle solmulle, jolla siitä ei vielä ole kopioita [VB00]. Näin viesti ennen pitkää saavuttaa vastaanottajan.

Epideeminen reititys on protokollana erittäin raskas. Jokainen viesti leviää verkon jokaiseen solmuun ja kuluttaa siten sekä lähetystehoa että talletustilaa. Epideeminen reititys toimittaa toisaalta viestin perille minimiajassa, koska se hyödyntää kaikki mahdolliset yhteydet solmujen väleillä. Tämän optimisuorituksen hintana on suurin mahdollinen yleisrasite. Satunnaisreitityksessä (engl. probabilistic routing, random routing) ja sen johdannaisissa tingitään viestin toimitusajasta vähentämällä viestistä tehtävien kopioiden määrää jollakin heuristiikalla. Suurissa verkoissa kumpikin näistä yksinkertaisista protokollista on edelleen liian raskas verkonkäytön ja muistinkulutuksen suhteen [JLW05].

DTN-verkkojen viestinvälityksen keventämiseksi on tutkittu myös erilaisia viestien redundanttiin koodaukseen perustuvia menetelmiä. Niissä alkuperäinen viesti koodataan

useaksi itsenäiseksi koodilohkoksi, jotka lähetetään erikseen verkkoon. Vastaanottaja pystyy rekonstruoimaan alkuperäisen viestin saatuaan riittävän määrän näitä koodilohkoja. Menetelmän tehokkuus perustuu siihen, että mitkä tahansa koodilohkot riittävät viestin koostamiseen eikä yksittäisen lohkon katoaminen matkalla estä viestin perillemenoa. Voidaan osoittaa, että pienellä yleisrasituksella voidaan taata viestin perillemeno hyvin suurella todennäköisyydellä [WJMF05, WB05].

Jos verkon topologiaa voidaan ennustaa tai mallintaa solmujen keskenään vaihtaman tiedon perusteella edes osittain, voidaan reititystä edelleen keventää [JFP04, BGJL06]. Suuri osa DTN-tutkimusta yrittääkin löytää tehokkaampia ja kevyempiä tapoja reitittää viestit lähettäjältä vastaanottajalle. Reitityksen ongelmia ja löydettyjä ratkaisumalleja käsitellään erikseen luvussa 3.

2.4 Kuljetuskerroksen toiminta

Reititys- ja verkkokerrokset yrittävät saada viestit perille lähettäjältä vastaanottajalle. Sen lisäksi tarvitaan sovelluksien välistä keskustelua eli viestikuittauksien saamista takaisin alkuperäiselle lähettäjälle. Näin lähettäjä- ja vastaanottajasovelluksien välille syntyy kuljetuskerroksen yhteys sekä saadaan keinoja hallita verkon ruuhkautumista.

Verkon kuormitusta voidaan rajoitetusti ohjata sovelluksista käsin. Välitettävät viestit voidaan luokitella kolmeen palveluluokkaan (engl. Quality of Service, QoS). Alimpaan bulkkiluokkaan (engl. bulk bundles) kuuluvat viestit saa toimittaa perille mahdollisimman pienellä resurssikulutuksella. Niitä korkeammalla olevan normaaliluokan viestit (engl. normal class bundles) toimitetaan niitä aikaisemmin. Pikaviestit (engl. expedited bundles) toimitetaan aina ensimmäisenä.

Luokat kertovat missä järjestyksessä saman solmun lähettämiä viestejä tulee käsitellä. DTN-verkon solmut voivat itse päättää, millä tavoilla ne toisten solmujen liikennettä välittävät eli ne voivat lähettää omat normaaliluokan viestinsä ennen toisten pikaviestejä. Internetin palvelunlaatumallissa eri kiireellisyysluokilla on globaali järjestys eli korkean prioriteetin paketti käsitellään ensimmäisenä jokaisessa välityssolmussa [BBCD98, BCS94].

Kuljetus- ja istuntokerrosten toiminta viivesietoisissa verkoissa on monimutkaista siksi, että viestinippuprotokolla peittää alleen erilaisten kuljetuskerrosten ominaisuudet. Esimerkiksi ruuhkanhallinta sekä viestien solmuista vievän tilan hallinta ovat vielä avoimia tutkimusongelmia [HA06, ZZ07]. Kummassakin näistä on kyse siitä, millä tavoilla solmut lähettävät toisilleen kuittauksia ja tietävät, mitä viestejä ne voivat poistaa muististaan.

Viestikuittaukset voidaan toteuttaa kahdella tavalla. Lähetetyistä viesteistä voidaan saada

kuittaus joko seuraavalta välittäjäsolmulta (engl. hop by hop) tai lopulliselta vastaanottajalta (engl. end to end). Kuittaus jokaiselta solmuväliltä kertoo, että viesti on edennyt seuraavalle solmulle, mutta se ei luonnollisestikaan takaa, että viesti koskaan päätyy varsinaiselle vastaanottajalle. Tämän takaamiseksi tarvitaan päästä päähän -kuittaus.

Yksinkertaisin tapa toteuttaa kuittausviesti on aktiivikuittaus (engl. active receipt), jossa kuittausviesti reititetään verkon läpi vastaanottajalta lähettäjälle aivan samaan tapaan, kuin alkuperäinenkin viesti. Tämä selvästikin toimii, mutta sen haittapuolena on verkossa liikkuvien ja mahdollisesti verkkoon jäävien viestien määrän kaksinkertaistuminen [HA06].

Passiivikuittauksessa (engl. passive receipt) säästetään resursseja tinkimällä kuittauksen perillemenoon kuluvasta ajasta. Vastaanottaja lähettää verkkoon kuittausviestin jokaiselle solmulle, joka vielä yrittää lähettää alkuperäistä viestiä eli ylimääräistä kopiota viestistä, jonka vastaanottaja on jo saanut. Tämä siivousviesti poistaa alkuperäisen viestin verkosta ja päätyy lopulta alkuperäiselle lähettäjälle, joka näin saa tiedon viestin perillemenosta [HABR05, HA06].

Kumpikin näistä kuittausprotokollista perustuu paljon resursseja käyttävään epideemisen reititykseen. Kuittaukset kuitenkin auttavat solmujen talletustilan hallintaa, joten niitä halutaan ehkä käyttää resurssinkulutuksesta huolimatta [HA06, Fall03].

2.5 Verkon toiminnan turvaaminen

Viestien aktiivinen käsitteleminen ja tallettaminen verkon solmuihin luovat DTN-verkkoihin erityisiä tietoturvaongelmia. Verkon solmut ovat mahdollisesti hyvin rajoittuneita suoritinteholtaan, jolloin käytettävien turvamekanismien tulee olla mahdollisimman kevyitä. Itse verkossa kulkevan liikenteen suojaus voidaan tehdä tavanomaisilla keinoilla, kuten Transport Layer Security (TLS), mutta solmujen pitää myös pystyä suojelemaan itseään ja verkkoa liialliselta resurssien kulutukselta kuten palvelunestohyökkäyksiltä ja väärennöksiltä [FSW07, RBF07].

DTN-sovellusten luottamuksellisuutta voidaan varmentaa sekä päästä päähän että solmuväli kerrallaan tai näiden eri yhdistelminä [KZH07]. Ehdotetussa turvallisuusmallissa määritellään erikseen viestin lähettäjä, vastaanottaja, lähde ja kohde (engl. sender, receiver, source, destination). Nämä voivat pareittain käyttää eri turvalähteitä esimerkiksi yhden solmuvälin viestinvaihdon suojaamiseksi eri menetelmällä kuin mitä lähettäjän ja vastaanottajan välillä käytetään.

Suurin käytännön ongelma on DTN-verkon luottamussuhteiden rakentaminen ja avainhallinta. Kaikki nykyiset julkisen avaimen järjestelmät perustuvat jatkuvasti saatavilla olevaan avainhallintapalveluun, jota osittuneesta verkossa ei välttämättä ole olemassa [FSW07, AKG+07]. Perinteisiä sertifikaattipohjaisia suojausratkaisuja kevyempi tapa todentaa viestien oikeellisuus on yksilötunnistepohjaisten (engl. identity-based cryptography, IBC) turvamenetelmien käyttö. DTN-verkoissa niitä käytetään todentamaan lähettäjä ajoittain vaihtuvalla verkon avaingeneraattorilta (engl. private key generator, PKG) saatavalla varmenteella. Yksilötunnistepohjaiset menetelmät sopivat käytettäviksi myös ilman jatkuvaa verkkoyhteyttä varmennepalveluun, sillä verkkoon levitetään lähettäjien varmennettu avain samalla kertaa itse viestin kanssa [AKG+07, KZH07]. Vastaanottaja todentaa saamansa viestin tuoreella lähettäjän tunnistavalla avaimella.

Viestien luotettavuuden lisäksi osassa DTN-verkkojen sovelluksista on toivottavaa taata lähettäjän anonymiteetti [KZH07, PFH04]. Tähänkin voidaan käyttää yksilötunnistepohjaisia turvamekanismeja, kunhan yhteyden päässä olevat solmut käyttävät lähettäjästä ja vastaanottajasta sopivaa pseudonyymiä. Välittävät solmut näkevät vain tämän pseudonyymin olemassaolon, mutta eivät voi yhdistää sitä todelliseen lähettäjään ja vastaanottajaan. Internetin sipulireititykseen perustuvia anonymisointipalveluita ei voida käyttää DTN-verkoissa [DMS04].

DTN-liikenteen välittäminen edellyttää sopimusta tai suostumusta verkon solmujen välillä. Silloinkin välittävät solmut saavat tarpeen tullessa poistaa ongelmaviestejä verkosta. Itse verkko voi siten suojautua välitettävän liikenteen haittavaikutuksilta. Perinteinen Internet ottaa välitettäväkseen kaiken sinne syötetyn liikenteen, joten verkko ei voi itse suojautua väärinkäytöksiltä [CBD+07, FSW07, SB07].

Suojamekanismien tarkoitus on estää luvattomien sovelluksien ja solmujen pääsy verkkoon sekä tunnistaa ja eristää vahingolliseksi osoittautunut solmu. Raskaiden turvamekanismien käyttö ei ole resurssinkulutuksen vuoksi toivottavaa, vaan verkossa voidaan sietää kohtuullisessa määrin tunnistamatonta liikennettä. Internetin kokemusten perusteella ei ole perusteltua olettaa, että mikään verkko olisi täysin suojassa haittaliikenteeltä [FC06].

Palvelunestohyökkäysten torjumiseksi on ehdotettu tuoreiden evästeiden käyttöä hyökkäysliikenteen pudottamiseksi verkosta [RBF07]. Ulkopuolisen hyökkäyksen havaitessaan solmut hyväksyvät välitettäviksi vain tuoreita, tunnistettuja viestejä. Myös DTN-verkkojen liikenteenhallinta- ja turvamekanismeja vastaan voidaan hyökätä tai käyttää niitä hyväksi. Arkkitehtuurikuvaukseen onkin jo täsmennetty, että Bundle-tason kuittausviestin käsittely ei saa tuottaa toista kuittausviestiä. Tämän mekanismin väärinkäytöllä voitaisiin monistaa liikennettä ja häiritä koko verkon toimintaa. Muidenkin kuittausvies-

tien hyödyllisyys voi jäädä niistä syntyvien turvaongelmien varjoon [FC06].

Viestinvälityksen fragmentointimekanismiin sisältyvät samat ongelmat osaviestien allekirjoittamisesta ja tunnistamisesta, kuin IP-protokollankin fragmenttien tapauksessa. Verkon reitityksessä on mahdotonta tunnistaa ja poistaa viestejä, joissa ei ole mukana kaikkia tarpeellisia tunnisteita. Kriittisten DTN-verkkojen tapauksessa on yksinkertaisinta olla käyttämättä fragmentteja [FC06].

DTN-verkkojen turvaongelmien ratkaisuissa on yritetty oppia Internetissä tehdyistä virheistä, eikä esimerkiksi käyttää viestinumeroina tasaisesti kasvavia helposti arvattavia laskureita [RBF07, FSW07]. Turvamenetelmien pitää olla tarpeeksi yksinkertaisia, että niitä käytetään mahdollisimman paljon ja jotta ne voidaan toteuttaa turvallisesti. Protokollien turvallisuuden lisäksi on tärkeä ottaa huomioon toteutuksien turvallisuus, sillä suuri osa Internetin tietoturvaongelmista johtuu huonosti tehdystä toteutuksista. Myös dokumentaation merkitystä korostetaan, jotta esimerkiksi turvattoman sensoriverkkototeutuksen käyttöönottaja tietää laitteistonsa rajat, eikä oleta siltä liian suurta turvatasoa [FC06].

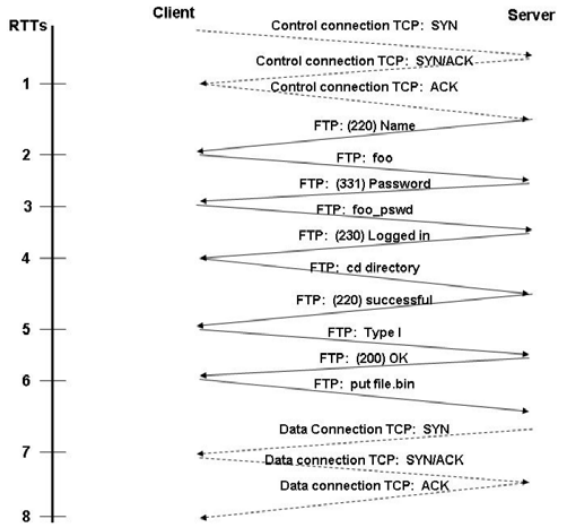
DTN-verkot tarvitsevat viestien käsittelyssä sekä solmujen ja niissä toimivien sovellusten toiminnan kohdentamisessa luotettavaa aikasynkronointia. Viestit tunnistetaan niiden luontiajan ja lähettäjän tunnisteen (EID) perusteella. Myös viestien vanhentamisessa eli vanhentuneiden viestien poistossa verkosta tarvitaan kaikille solmuille riittävän yhtenäistä aikatietoa [CBD+07, SB07]. Aikasynkronoinnin toteutusta ei ole erikseen määritelty, joten siihen ajatellaan käytettäväksi DTN-arkkitehtuurin ulkopuolella olevia, kuhunkin sovellusalueeseen soveltuvia keinoja.

Reitityksessä tarvitaan aikasynkronointia esimerkiksi aikataulunmukaisten tai ennustettujen kontaktien käyttämisessä. Torkkuvien solmujen pitää herätä ajoissa ollakseen valmiina hyödyntämään tulevaa kontaktia. Vastaavalla tavalla sovellukset rekisteröityvät ottamaan vastaan tietyn EID-osoitteen viestejä tietyksi ajaksi. Tämä rekisteröinti vastaa ryhmälähetyksen vastaanottorekisteröintiä, mutta se tehdään vain tietyksi ajaksi. Määräajan voi ilmaista sovellus itse, muuten käytetään solmun tai sovelluksen oletusarvoja.

2.6 DTN-verkkojen sovellukset

Nykyiset verkkosovellukset on rakennettu luotettavan päästä päähän -yhteyden varaan. Niiden arkkitehtuurissa sovelluskerroksella ei ole rajapintaan kuuluvaa mahdollisuutta vaikuttaa alla olevan verkkoyhteyden toimintaan tai saada siitä tietoa.

DTN-arkkitehtuurimallissa sovelluksille annetaan mahdollisuus ottaa kantaa verkon toimintaan ja resurssikulutukseen. Verkkoa ei siis yritetä piilottaa sovellukselta tai sen käyt-



Kuva 4: FTP tarvitsee kahdeksan kiertoviiveen verran TCP-segmenttien vaihtoa asiakkaan ja palvelimen välissä ennen tiedostonsiirron alkua [FD06].

täjältä, vaan sovelluksen kuuluu huomioida mahdollisten yhteyskatkojen ja toimitusviiveiden vaikutukset [DBF04, Ott06]. Viivesietoisten verkkojen arkkitehtuuri poistaa jatkuvan yhteyden (engl. always on) vaatimuksen verkkoa käyttäviltä sovelluksilta ja korvaa sen parhaalla saatavilla olevalla yhteydellä (engl. always best connected).

Sovelluksilta myös edellytetään vastuun ottamista verkon palvelutason ylläpitämisestä. Niiden tulee esimerkiksi välttää monivaiheista osapuolten välistä neuvottelua etenkin silloin, kun kiertoviive on suuri. Esimerkiksi kuvassa 4 esitetty Internetin tiedostonsiirtoprotokolla FTP vaatii kahdeksan kiertoviiveen verran TCP-viestien vaihtoa ennen kuin tiedoston siirto voi alkaa [FD06]. Suurten siirtoviiveiden tapauksessa tähän tuhlaantuu paljon aikaa.

Sovellusten pitää selvitä yhteyshäiriöistä ja laitteiden uudelleenkäynnistyksistä silloinkin, kun viestinvälitys on vielä kesken. Perinteiset Internet-verkkosovellukset pysähtyvät käyttäjän kannalta epätyydyttävällä tavalla, jos verkkoyhteys katkeaa kesken toiminnon. Lisäksi sovellusten tulisi vuorostaan kertoa verkolle välitettäväksi annettujen viestien voimassaoloajasta ja tärkeydestä. Näin solmut voivat toimia tarkoituksenmukaisemmin ja pudottaa verkosta resurssien loppuessa vähemmän tärkeitä tai vanhentuneita viestejä.

Viivesietoisten verkkojen ominaisuudet tulevat esille eri sovelluksissa eri tavoilla. Ennakoimattomat yhteyskatkot ja yhteysmahdollisuudet kuvaavat esimerkiksi ajoneuvoverkkojen tai liikkuvien käyttäjien verkkojen haasteita. Internetiä suurempia toimitusviiveitä esiintyy siirrettäessä tietoa fyysisillä talletusvälineillä, avaruustutkimuksessa tai meren-

tutkimuksessa käytetyillä akustisilla langattomilla yhteyksillä.

Turvaominaisuudet sekä palvelunlaatuluokitukset ovat tärkeitä sotilas- ja pelastuskäytössä, joissa verkon suojaaminen ulkoiselta häirinnältä on ensiarvoista. Sensoriverkkojen solmujen taas tulee tulla toimeen mahdollisimman pienillä resursseilla, jolloin verkkoyhteyksien määrän minimointi on tärkeää.

Osa verkoista on haasteellisia puhtaasti käytännöllisistä syistä. Esimerkiksi langattoman tietoliikenteen häiriöitä voidaan vähentää ja kantamaa lisätä yksinkertaisesti käyttämällä suurempaa lähetystehoa. Suurempi tehonkäyttö johtaa kuitenkin joko suurempaan energialähteeseen tai lyhyempään toiminta-aikaan, joista kumpikin saattaa nostaa toiminnan kokonaiskuluja merkittävästi. DTN-sovelluksissa halutaankin käytännössä tulla toimeen mahdollisimman niukoilla resursseilla mahdollisimman pitkään.

Haasteellisissa verkoissa on käytetty sovelluksia jo ennen DTN-tutkimuksen syntyä, joten olemassa olevia DTN-toteutuksia on vielä vähän. Seuraavassa käydään läpi sovellusalueita, joiden toteuttamisessa tullaan suurella todennäköisyydellä käyttämään DTN-arkkitehtuurin soveltuvia osia.

Alueet Internetin ulkopuolella

DTN-mallin mukaisilla sovelluksilla voidaan toteuttaa yhteyksiä ja viestinvälitystä alueille, joissa ei ole käytettävissä perinteistä verkkoinfrastruktuuria. Kehittyvien maiden hajaasutusalueilla voidaan hyödyntää aurinkopaneeleilla tuotettua sähköä esimerkiksi koulujen ja paikallishallinnon tietokoneiden käyttämiseksi, mutta verkkoyhteyksien luominen saattaa olla todella kallista. Sähköposti, digitaaliset äänikirjeet ja valittujen WWW-sivujen ennaltanouto voivat kuitenkin toimia myös ilman reaaliaikaista yhteyttä [OK06]. Käyttämällä kyliä kiertävää paikallisbussia viestien välittämiseen voidaan tarjota näitä peruspalveluita suurille käyttäjämäärille hyvin edullisesti [DBF04, PFH04].

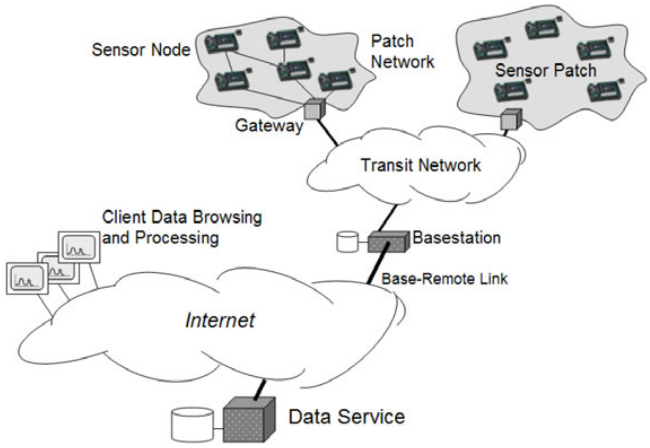
Sähköisessä puhepostissa (engl. VoiceMail Over IP) käyttäjä tallentaa tietokoneella ääni- tai videoviestin, joka toimitetaan verkon kautta vastaanottajalle. Viestien lähetys ja vastaanotto on mahdollista myös luku- ja kirjoitustaidottomille, sillä palvelua voidaan käyttää avustajan kanssa. Myös viranomaispalveluihin kehitetään tällaisia tietokonetaitoisen välittäjähenkilön avulla käytettäviä sovelluksia (engl. mediated user interface) [BDH+06]. Jokainen uusi viestintätapa, vaikkakin Internetin käyttöön tottuneiden mielestä kömpelö ja vaatimaton, lisää näillä seuduilla asukkaiden hyvinvointia ja tukee esimerkiksi vapaampaa tiedonvälitystä, tehokkaampaa kouluopetusta ja muutakin myönteistä kulttuurikehitystä [BDH+06].

Viestintäinfrastruktuuri voi puuttua joltakin alueelta myös siksi, että alue on joutunut

luonnononnettomuuden tai sotilastoimien kohteeksi. Pelastus- ja taistelutoimien operatiivisten tukijärjestelmien tulee toimia itsenäisesti verkottumalla häiriöistä ja katkoista huolimatta. Suora henkilöiden välinen kommunikointi voi tapahtua tavalliseen tapaan radiopuhelimilla, mutta esimerkiksi karttojen, kuvien ja tilannetietojen välittämiseen tarvitaan omia hajautettuja sovelluksiaan [MED07].

Kummassakin näistä käyttötapauksista voi olla käytettävissä rinnalla myös muita viestintätapoja, esimerkiksi satelliittiyhteyksiä, jotka saattavat olla kaistanleveydeltään pienempiä tai kalliimpia, mutta joita silti halutaan käyttää esimerkiksi kontrolliliikenteen välittämiseen. DTN-sovellusarkkitehtuuri osaa valita kulloisenkin viestintätarpeen mukaan sopivimman yhteystavan.

Alueet lähellä maapallon maantieteellisiä napoja ovat geostationaaristen satelliittien varaan rakentuvien yhteyksien katvealueilla, joten siellä DTN-tekniikat soveltuvat hyvin yhteyksien toteuttamiseen. Viestilauttana voidaan käyttää sopivaa huoltoliikenteen mukana kuljetettavaa fyysistä talletusvälinettä [SAP].



Kuva 5: Sensoriverkkojen keräämät tiedot välitetään yhdyskäytävien ja välittäjäverkkojen kautta Internetiin ja tutkijaryhmän tietovarastoon [MCP+02].

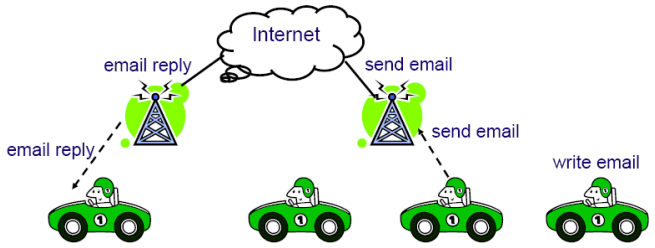
Sensoriverkot

Sensoriverkoilla tutkitaan erilaisia bio- ja geotieteiden tutkimusongelmia tekemällä mittauksia useissa pisteissä tietyllä maantieteellisellä alueella. Tutkimusalueelle levitetään jollakin tekniikalla pieniä, yleensä paristoilla toimivia mittalaitteita eli sensoreita. Nämä sensorit keräävät haluttuja mittaustietoja ja lähettävät ne kuvan 5 mukaisesti keräysasemalle, joka on mahdollisesti yhteydessä Internetiin ja sitä kautta tutkimusryhmään [LGE06, MCP+02, HM+06, WW07]. Langaton yhteys voi käyttää myös akustista kantoaaltoa, jolloin etenemisviive on tarpeen ottaa huomioon viestintää suunniteltaessa [SH03, HYW+06].

Sensorit voivat olla myös liikkuvia, esimerkiksi eläimiin kiinnitettyjä laitteita, joiden avulla kerätään tietoa sekä eläinten liikkeistä ja ryhmäytymisestä [JOW+02] että niiden elinympäristöstä [SH03, MCP+02].

Mittaustuloksia halutaan yleensä kerätä mahdollisimman suurelta alueelta ja pitkältä ajalta, joten sensorit toimivat hyvin niukoilla energia- ja talletusresursseilla. Pieni lähetysteho säästää energiaa, mutta on haasteellisissa ympäristössä, esimerkiksi jäätikön sisään poratuissa sensoreissa, altis hyvin voimakkaalle vaimenemiselle ja häiriöille [HM+06].

DTN-protokollat pystyvät toimimaan tällaisissa oloissa tehokkaasti ja välittämään mittaustiedot koko sensoriverkon läpi yksi askel kerrallaan. Lähetyksien ja tarvittavan prosessoinnin määrää halutaan näissä sovelluksissa minimoida, joten reitityksessä ja talletuksessa käytetään mahdollisimman keveitä algoritmeja [WDW07]. Yhtenäinen tietoliikenne- ja sovellusrajapinta antaa myös mahdollisuuden käyttää samoja DTN-viestilauttoja purkamaan useamman sensoriverkon tietoja samalla kertaa.

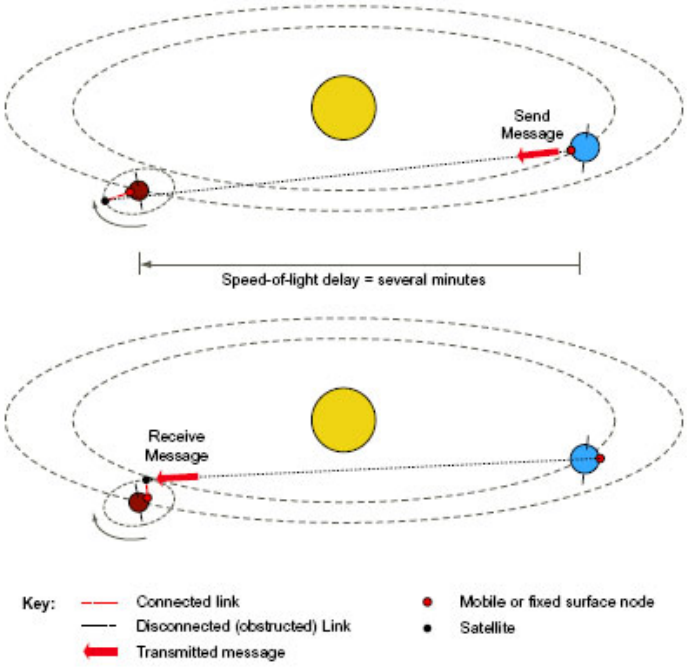


Kuva 6: Nomadisen käyttäjän pääsy sähköpostipalveluun voi perustua kullakin hetkellä erilaisiin verkkoyhteyksiin [Spy07].

Mobiilit ad hoc -verkot

Nomadisten käyttäjien ja ajoneuvojen välisiä yhteyksiä luonnehtivat usein esiintyvät yhteyskatkot, vaihtelevat yhteystekniikat sekä ennustamattomat toisten solmujen kohtaamiset. Niissä toimivien sovellusten toteuttamiseen sopivat hyvin DTN-arkkitehtuurin toimintamallit opportunistisista yhteyksistä sekä viestejä kuljettavista verkon solmuista. Näin voidaan toteuttaa edullisempia yhteyksiä valituille sovelluksille vähemmällä infrastruktuurilla. Kuvassa 6 lähetetään ja vastaanotetaan sähköpostia käyttämällä kullakin hetkellä saatavilla olevaa yhteyttä sopivaan langattomaan verkkoon [HKL+07].

Näissä sovelluksissa langattomilla yhteyksillä varustetut henkilökohtaiset laitteet verkostoituvat keskenään ja täydentävät näin infrastruktuuriverkon palveluita toimimalla toisilleen tukiasemina. Nykyisillä nopeilla langattomien verkkojen tekniikoilla saadaan lyhyilläkin kontakteilla siirrettyä merkittäviä määriä tietoa [OK04].



Kuva 7: Yhteyksissä toisille planeetoille on otettava huomioon signaalin etenemisviive [War03].

Pitkät etäisyydet

Muiden planeettojen tutkimus on edelleen vilkastumassa, joten myös niillä toimivien laitteiden tietoliikenneyhteyksissä haluttaisiin käyttää yhteisiä standardiprotokollia ja tietoliikennesatelliitteja. Tavalliset Internet-yhteydet eivät näissä sovelluksissa toimi valon nopeudesta johtuvan hyvin pitkän kiertoviiveen vuoksi. Internet-protokollilla on esimerkiksi mahdotonta välittää suoraan Marsissa olevan robotin webbikamerakuvaa, sillä TCP-yhteyttä sinne ei saada muodostettua ollenkaan [FCG06, WPP+07]. Kuvassa 7 on esitetty, miten signaalin etenemisviive vaikuttaa ajastettujen kontaktien käyttämiseen planeettojen välisessä viestinnässä.

Tällä hetkellä liikennöinti aurinkokunnassa käyttää *de facto* standardina tiedostonsiirron semantiikan tarjoavaa CFDP-protokollaa [CCS02], joten sovellustason viestinvälityksen semantiikalla toimivien DTN-protokollien käyttöönotto onnistuisi suhteellisen helposti.

Planeettojen välisillä yhteyksillä DTN-arkkitehtuurin toimitusvalvonnan siirrosta on selvää hyötyä siirrettäessä eheystarkistukset kaikkein virhealtteimmalle välille, kuten esimerkiksi Maan ja Marsin väliselle yhteydelle. Olisi resurssien tuhlaamista tehdä uudelleenlähetyksiä päästä päähän toisen planeetan pinnalla olevan mönkijän ja tutkimusryhmän tietovaraston välillä.

3 DTN-verkkojen reititys

Reititys perinteisissä verkoissa on periaatteessa yksinkertaista. Viesti annetaan sille välittäjälle, jolla on kaikkein edullisin, yleensä lyhyin, reitti vastaanottajan suuntaan. Koska yhteydet ovat luotettavia, riittää lähettää verkkoon vain yksi kopio viestistä. Viive- ja häiriösietoisten verkkojen reititys poikkeaa tästä merkittävästi yhteyskatkojen, solmujen rajoitetun talletustilan ja käytettävyyden sekä jatkuvan yhteyden puutteen vuoksi [LDS03, JFP04, Zha06].

DTN-verkkoihin esitettyjen reititysprotokollien suuri kirjo kuvaa sekä näiden verkkojen erilaisia sovellusalueita, että niitä erilaisia tapoja, joilla verkkoja voidaan teoreettisesti mallintaa [JFP04, SPR05, Zha06]. DTN-arkkitehtuurilla voidaan kuvata hyvin erilaisia haasteellisia verkkoja, joten reitityksen toteutustapa on valittava jokaisen sovellusalueen vaatimuksien mukaan.

Tietojenkäsittelytieteen teorian kannalta viivesietoiset verkot ovat ajallisesti muuttuvia monipolkuverkkoja (engl. time varying multigraph), joiden topologia ja jäsensolmujen joukko voivat muuttua usein [DBF04]. Tällaisten verkkojen käsittelyyn voidaan soveltuvin osin käyttää tavallisia menetelmiä, esimerkiksi Dijkstran lyhyimmän reitin etsivän algoritmin versiota, joka ottaa yhteysajat huomioon [JFP04]. Moniin käytännön sovelluksiin tarvitaan kuitenkin yksinkertaisempia ja vikasietoisempia verkkomalleja.

3.1 Verkkojen mallintaminen

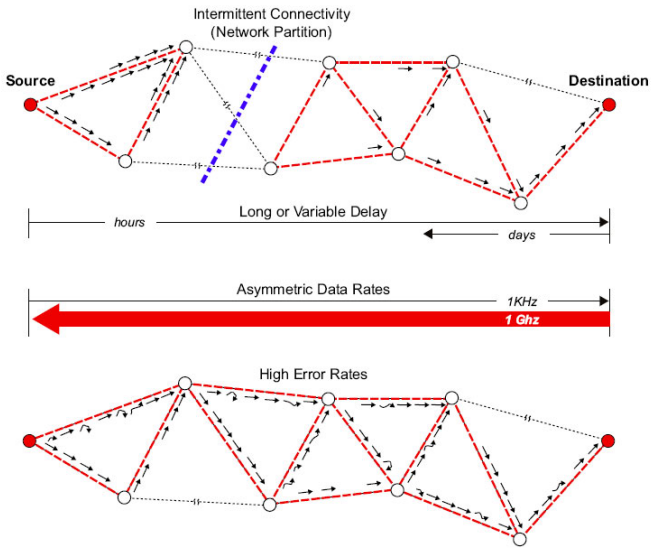
Ennustamattomia yhteyksiä on tutkittu ja mallinnettu itseorganisoituvia tilapäisverkkoja eli ad hoc -verkkoja varten [Per01, DRCY00]. Myös mobiiliverkkojen ja liikkuvien verkkojen (engl. network mobility, NEMO) ominaisuudet, kuten yhteyksien syntyminen ja katkeaminen, sekä solmun siirtyminen toisen tukiaseman kantoalueelle, kuvaavat joitakin DTN-verkkojen ominaisuuksista. Seuraavassa esitellään näitä langattomien ja ad hoc -verkkojen peruskäsitteitä.

Tiiviissä (engl. dense) langattomassa verkossa kaikki verkon solmut ovat toistensa kuuluvuusalueilla. Kaikki solmut voivat siis keskustella keskenään noudattamalla verkossa käytettävää linkkitason protokollaa, jossa lähettäjä selvittää vastaanottajan linkkiosoitteen ja lähettää sanomansa siihen. Vaikka IEEE 802.11 -infrastruktuuriverkoissa verkkokehys kiertää todellisuudessa tukiaseman kautta vastaanottajalle, lähettävän solmun kannalta se lähetetään suoraan vastaanottajan MAC-osoitteeseen (Medium Access Control, siirtokerroksen verkko-osoite) [IEE99].

Harvan (engl. sparse) verkon kaikki solmut eivät ole toistensa kuuluvuusalueella. Ne ovat kuitenkin saman verkon jäseniä eli esimerkiksi käyttävät yksikäsitteisiä verkko-osoitteita. Tällaisessa verkossa liikennöinti verkon laidalta toiselle edellyttää viestin välittämistä solmuväli kerrallaan lähettäjältä vastaanottajalle.

Tämä tuo esille uusia vaatimuksia sekä verkon jäsensolmuille, että verkossa käytettävälle tiedonsiirtoprotokollalle. Tällaisessa verkossa solmujen on osallistuttava aktiivisesti toistensa liikenteen välittämiseen. Niiden on otettava vastaan verkkopaketteja, jotka ovat menossa jollekin muulle solmulle, pidettävä niitä itsellään jonkin aikaa ja lähetettävä eteenpäin, kun mahdollisuus siihen ilmaantuu (engl. store-and-forward).

Osittunut (engl. partitioned) verkko on jakautunut osiin niin, että kaikkien solmujen välillä ei voida välittää paketteja. Osaverkkojen solmut eivät tästä yleensä tiedä ja olettavat verkon olevan edelleen ehyt. Tällaisessa verkossa osaverkkojen reunoilla olevat solmut joutuvat tallettamaan toisiin osaverkkoihin meneviä viestejä mahdollisesti pitkäksikin aikaa.



Kuva 8: DTN-verkoissa esiintyy yhteyskatkoja ja osittumista (engl. intermittent connectivity, network partition), pitkiä siirtoviiveitä (engl. delay), epäsymmetrisiä siirtoteitä (engl. asymmetric data rates) sekä suuria virhetiheyksiä (engl. high error rates) [War03].

Jos osittumisen ja yhteyskatkojen katsotaan kuuluvan verkon ominaisuuksiin eli verkko on vain ajoittain kytketty (engl. intermittently connected), ja verkkoa käyttävät sovellukset ovat mukautuneet katkoihin, käytetään termejä viivesietoinen verkko ja häiriösietoinen verkko (engl. Delay/Disruption Tolerant Network, DTN). Nykyään DTN onkin yleisnimi tällaisille haasteellisille verkoille. Kuva 8 esittää DTN-verkkojen ominaispiirteet, kuten

osittumisen, pitkät siirtoviiveet, suuret virhetiheydet sekä epäsymmetrisen siirtotien.

DTN-verkkojen sovellukset voivat toimia huomattavan tehottomissa verkko-olosuhteissa tai jopa kokonaan ilman yhteyksiä [OKD06]. Verkossa liikkuvan tiedon määrä ja yhteyksien saatavuus vaikuttavat toki verkkoyhteyksien laatuun, mutta tällaisiin oloihin kehitetyt sovellukset eivät lamaannu yhteyden puuttumiseen.

Reititys

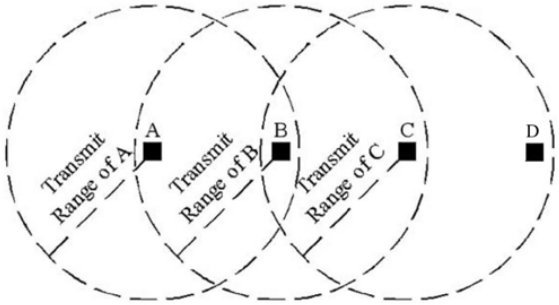
Jos DTN-verkon solmut eivät ole osoitettavissa suoraan linkkikerroksella, tarvitaan tapa löytää reitti verkon läpi lähettäjältä vastaanottajalle. Tilapäisverkkojen reititystiedon muodostamiseen on kaksi lähestymistapaa, joista kumpikin edellyttää verkon olevan tarpeeksi tiivis [DRCY00, TZZ03]. Perinteisten verkkojen tapaan voidaan toimia sen varassa, että solmut lähettävät tiedot tietämistään reiteistä toisilleen. Näin jokaisella solmulla on käsitys verkon topologiasta ja käytettävissä olevista yhteyksistä. Esimerkiksi Optimized Link-State Routing (OLSR) -reititysprotokolla toimii paremmin verkoissa, joissa solmujen väliset yhteydet eivät muutu usein [DRCY00].

Toinen lähestymistapa on hankkia reititykseen tarvittava tieto vasta viestin lähetystarpeen ilmaannuttua. Lähettävä solmu lähettää verkkoon reititystietopyynnön, joka etsii reitin vastaanottajalle asti tai palauttaa tiedon siitä, että reittiä ei voida halutussa ajassa löytää [DRCY00]. Myös tällainen esimerkiksi Ad-hoc On Demand Distace Vector -protokollan (AODV) käyttämä toimintatapa on käyttökelpoinen vain tarpeeksi tiiviissä verkossa.

Myös reitin käyttäminen voi tapahtua kahdella tavalla. Lähdereitityksessä (engl. source routing) paketin lähettävä solmu muodostaa reitin oman verkkokäsityksensä perusteella, liittää reitin pakettiin ja lähettää sen ensimmäiselle välittäjäsolmulle. Paketti kulkee verkon solmulta toiselle lähettäjän määräämää polkua. Paketti jää matkalle, jos verkon topologia muuttuu niin, että jotain tiettyä yhteysväliä ei enää ole olemassa. Lähdereititystä voidaan käyttää vain viestin toimitusajan puitteissa muuttumattomissa verkoissa. Hyppykohtaisessa reitityksessä (engl. per contact routing, hop by hop routing) jokainen välittävä solmu tekee reitityspäätöksen sen hetkisen verkkotopologian mukaan. Paketti annetaan välitettäväksi sille solmulle, jolla on sillä hetkellä paras reitti vastaanottajan suuntaan.

Osa DTN-verkoista edellyttää solmujen liikkuvan toistensa suhteen ja kuljettavan mukanaan viestejä toisille solmuille (engl. store-carry-and forward, mobility assisted routing/forwarding). Näissä tapauksissa solmujen liikkuminen on olennainen osa viestien reititystä eli solmut toimivat viestien siirtokerroksena. DTN-arkkitehtuurissa verkon viestejä voidaan kuljettaa myös erillisillä fyysisillä tietovälineillä [CBD+07].

Lähettäjä voi edellyttää viestin välitalletusta tavalla, joka takaa sen säilymisen myös välittäjäsolmun uudelleenkäynnistyksien yli (engl. custody transfer). Tällöin viestin välityksestä kulloinkin vastaava solmu huolehtii sen lähettämisestä edelleen ja edellinen solmu voi poistaa viestin muististaan. Tämän ansiosta uudelleenlähetys voidaan siirtää koko yhteyden virhealtteimmalle välille, eikä turhaan tuhlata lähettäjän ja vastaanottajan resursseja yrittämällä lähettää viestiä koko verkon läpi.



Kuva 9: Piilolähettäjäongelmassa C voi katkaista A:n lähetyksen B:lle, koska se ei kuule A:n lähetystä. Julkilähettäjäongelmassa C pidättäytyy turhaan lähettämästä D:lle, koska se kuulee B:n lähetyksen A:lle [HD03].

Siirtotie

Kaikkien langattomien verkkojen tietoliikennettä haittaavat jaetusta siirtotiestä ja radioaaltojen kuuluvuudesta johtuvat kuvassa 9 esitetyt erityisongelmat. Niistä tärkein on piilolähettäjäongelma (engl. hidden terminal problem), jossa solmun A lähetys solmuun B katkeaa solmun C aiheuttamaan törmäykseen (engl. collision), koska C ei havaitse solmun A kantoaaltoa. Toinen suorituskykyä laskeva ongelma on julkilähettäjäongelma (engl. exposed terminal problem), jossa verkon solmu C turhaan pidättäytyy lähettämästä kauempana olevalle vastaanottajalle D, koska se havaitsee solmun B lähetteen solmulle A. Todellisuudessa solmun C lähetys ei häiritsisi solmun A vastaanottoa [XS02].

Sovelluksen verkkoliikennöinnin eri suuntaan etenevien komponenttien keskinäinen häirintä on merkittävä käytännön ongelma monihyppyisissä langattomissa verkoissa. Tällöin esimerkiksi TCP-yhteyden kuittauspaketit kilpailevat välittäjäsolmuissa lähetysvuorosta saman yhteyden hyötykuormapakettien kanssa. Tämän keskinäisen häirinnän vuoksi TCP-protokollan suorituskyky romahtaa monihyppyisissä verkoissa [MRPS04, PWWM05]. Sama ongelma esiintyy myös viivesietoisissa verkoissa, sillä niissäkin kuittausten ja vuonvalvonnan toteutus edellyttää vastakkaisiin suuntiin kulkevia viestivirtoja. Näiden viestien aikaskaala on erilainen kuin TCP-protokollassa, joten ongelma ei ole niin suuri.

Viive

DTN-verkkojen reitityksen tehokkuuden mittaamiseen on kehitetty erilaisia suureita. Lähetetyn liikenteen suhde vastaanotettuun, verkkolähetyksien määrä tai tehonkulutus ovat tärkeitä mittareita, mutta sovelluksien kannalta merkittävin on viestin toimitusviive. Viive jakautuu odotus-, jonotus-, lähetys- ja etenemisviiveisiin [JFP04] . Nämä ovat toisistaan riippumattomia tekijöitä ja vaikuttavat kaikki osaltaan kokonaisviiveeseen.

Odotusviive on aika, joka viestin pitää odottaa siitä lähtien, kun se saapuu solmuun aina siihen asti, kunnes seuraava välittäjäsolmu on käytettävissä. Odotusviiveen pituus riippuu solmujen liikkeistä ja yhteysaikatauluista sekä viestin saapumishetkestä.

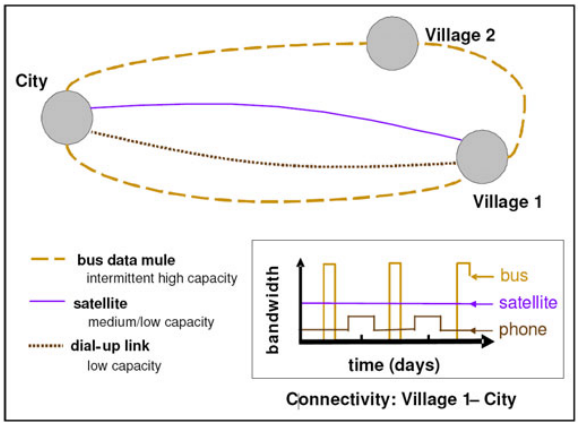
Jonotusviive on aika, joka solmulta kuluu viestin edellä jonossa olevien viestien lähettämiseen. Edellä jonossa ovat sekä kaikki korkeamman prioriteetin viestit, että aikaisemmin saapuneet saman prioriteetin viestit. DTN-verkoissa prioriteetit ovat lähettäjäkohtaisia, joten solmun omat normaalin prioriteettiluokan viestit voivat ohittaa välitettävät toisten solmujen pikaviestit. Jonotusviive riippuu yhteyden nopeudesta ja muun kilpailevan liikenteen määrästä.

Lähetysviive on aika, joka kuluu viestin lähettämisessä verkkoon. Tämä viive saadaan suoraan viestin koosta ja yhteyden nopeudesta. Etenemisviive kertoo, miten kauan viestiltä menee saavuttaa seuraava solmu. Viiveen suuruus riippuu käytetystä siirtotekniikasta sekä välimatkasta. Viivesietoisissa verkoissa voidaan käyttää perinteisten radioaaltojen ja optisten yhteyksien lisäksi myös akustisia tai fyysisiä siirtoteitä esimerkiksi kuljettamalla tietovälinettä paikasta toiseen. Tällöin siirtoviive voi olla suurin kokonaisviiveen komponentti.

Reititysprotokollia vertaillaan lähinnä kokonaisviiveen minimin tai sen odotusarvon perusteella. Protokollan valinta voi vaikuttaa vain jonotus- ja lähetysviiveisiin, lukuun ottamatta niitä protokollia, joissa solmun liikkumista ohjataan helpottamaan yhteyksien ottamista. Näissä protokollissa voidaan siis pienentää solmun aktiivisilla toimilla myös odotusviivettä [LR03].

3.2 Kontaktit

Perinteisissä verkoissa yhteyksien ajatellaan olevan nopeudeltaan kiinteitä ja käytettävissä koko ajan. Mahdollisten katkojen ajatellaan olevan lyhytaikaisia suhteessa normaalitoimintaan. Sen sijaan viivesietoisissa verkoissa varaudutaan pitkiin katkoihin sekä nopeudeltaan erilaisiin ja mahdollisesti vain hetkittäin toimiviin yhteyksiin.



Kuva 10: DTN-reititys ottaa huomioon erilaisten kontaktien kapasiteetin ja hinnan. Puhelinyhteys on edullisempi öisin, bussi kulkee kylän kautta harvoin, mutta kuljettaa paljon tietoa kerrallaan. Satelliittiyhteys on tarjolla koko ajan, mutta se on kallis [DBF04].

Verkon solmulle voi olla kuvan 10 esittämällä tavalla käytettävissä vaihtoehtoisia, nopeudeltaan, käyttöajaltaan tai kustannuksiltaan erilaisia tapoja saada viesti perille [DBF04, CBD+07]. Viestille voidaan sovellustasolla määritellä myös laatuluokka, jolloin kiireelliset pikaviestit voidaan lähettää haluttaessa kalliimmalla yhteystavalla.

DTN-verkkojen reitityksessä mallinnetaan solmujen välisten yhteystapahtumien eli kontaktien (engl. contact) ominaisuuksia. Perinteisen verkkoyhteyden suorituskykyä riittää kuvamaan yleensä vain yksi lukuarvo, yhteyden nopeus bitteinä sekunnissa. Kontaktin kuvaamiseen tarvitaan kaksi skalaariarvoa, yhteyden nopeus sekä sen kesto. Näiden tulona saadaan kontaktin tilavuus (engl. volume) eli sen aikana siirrettävän tiedon määrä. Kontaktin tiedonsiirtonopeus saattaa olla erilainen eri suuntiin tai muuttua yhteyden aikana esimerkiksi solmujen lähestyessä toisiaan, jolloin tilavuus saadaan integroimalla nopeusfunktio kontaktin kestoajan yli [JFP04].

Kontaktin muita ominaisuuksia voivat olla esimerkiksi kustannus, siirtoviive tai kerralla lähetettävän tiedon määrä. Verkon tai solmun kuormitus (engl. load) määritellään siirrettävän tiedon määrän suhteena kontaktien tilavuuteen. Jos kuormitus on alle yhden, ei ruuhkaantumista tapahdu [JFP04].

Ajastetut ja ennustetut kontaktit

Osassa DTN-verkkojen sovelluksista tulevat kontaktit tiedetään ennakolta. Esimerkiksi planeettojen asemat ja asennot voidaan laskea tarkasti, joten radioyhteyden päissä olevat solmut voivat lähettää ja vastaanottaa viestit täsmälleen oikeaan aikaan. Viestilauttana toimivalla bussillakin on aikataulu, mutta se ei ole aivan yhtä täsmällinen. Yhteyteen osal-

listuvien solmujen pitää siten varautua odottamaan yhteyden muodostumista. Suunnilleen tiedossa oleva aikataulu kuitenkin helpottaa solmujen muun toiminnan ja reitityksen suunnittelua.

Useissa käytännön sovelluksissa kontakteja ei tiedetä etukäteen. Tällöin pyritään löytämään verkon solmujen keskinäisistä kohtaamisista säännönmukaisuuksia, joita käytetään tulevien kontaktien ennustamiseen. Tätä käsitellään tarkemmin luvussa 3.3.

Niukoilla resursseilla toimittaessa ennuste tulevan kontaktin ajankohdaksi on parempi vaihtoehto, kuin joutua odottamaan yhteyttä koko ajan. Näin solmu voi esimerkiksi sammuttaa itsensä kokonaan tai mennä lepotilaan siihen asti, kunnes kontaktin aika lähestyy. Vaihtoehtona tälle solmun pitäisi koko ajan olla varautunut kontaktin käsittelyyn, mikä voisi merkittävästi kuluttaa rajoitettuja resursseja.

Ajastettujen ja ennustettujen kontaktien käyttäminen edellyttää toimivaa aikasynkronointia sekä, etenkin itsensä sammuttavissa solmuissa, riittävän tarkkaa reaaliaikakelloa. DTN-arkkitehtuuri ei ota kantaa aikasynkronoinnin käytännön toteuttamiseen [CBD+07], mutta normaalia NTP-protokollaa ei kuitenkaan voida käyttää aikapalvelimien mahdollisen saavuttamattomuuden vuoksi [MN04].

Opportunistiset kontaktit

Joissakin DTN-verkkojen käyttötapauksissa solmut kohtaavat toisensa ilman ennakkotietoa, reitityksen kannalta satunnaisesti, mutta sovelluksen kannalta onnekkaasti. Tyypillinen tällainen opportunistinen kontakti on kahden auton kohtaaminen tiellä. Yhteyden loppumista voidaan rajoitetusti ennustaa tarkkailemalla langattoman yhteyden signaalivoimakkuuden muuttumista [OK04], mutta silti yhteys voi katketa ilman ennakkovaroitusta. Tällöin voidaan siihen asti lähetetystä viestin osasta muodostaa oma fragmenttinsa ja lähettää loppuosa seuraavassa kontaktissa.

Opportunistiset kontaktit ovat reitityksen kannalta haastavimpia, sillä niiden toteutumiseen tai suorituskykyyn ei voida luottaa. Toisaalta riittävän tiiviissä verkossa niitä esiintyy suurella todennäköisyydellä niin usein, että niiden varaan voidaan perustaa ainakin vähemmän kriittisiä palveluita. Tilanne muistuttaa riittävän käyttäjämäärän saavuttanutta vertaisverkkoa, jossa minkään yksittäisen koneen ei tarvitse olla päällä kokonaispalvelun toimimiseksi.

Opportunistiset kontaktit ovat ongelmallisia myös verkon solmujen tehonkulutuksen kannalta. Solmun pitää olla riittävässä valmiudessa hyödyntää ilmaantuvaa kontaktia, mikä voi nopeasti kuluttaa sen energiavaroja. Erääksi ratkaisuksi tähän on ehdotettu kantoaallon tunnistamiseen tarkoitettua erittäin pienitehoista dedikoitua radiovastaanotinta [HYW+06].

Opportunistisen kontaktin ilmestyttyä tämä radiopiiri herättää solmun, joka tekee viestinvaihdot kontaktin aikana normaalilla radiollaan. Tällaisen erillisen radiopiirin hinta ja tehonkulutus voivat olla hyvin pieniä, jolloin siitä on merkittävää etua esimerkiksi viestilautan purkamissa sensoriverkoissa.

3.3 Kontaktien mallinnus

Siinä missä muissa verkoissa solmujen liikkuminen on ongelma, perustuu osassa DTN-verkoista viestinvälitys liikkuvien solmujen mukanaan kuljettamiin viesteihin (engl. store-carry-and-forward, mobility assisted routing/forwarding). Solmujen liikkuminen ja kohtaaminen toteuttavat verkon siirtokerroksen toimintaa, joten niiden kuvaus ja mallintaminen ovat tärkeä osa reitityksen tutkimusta [CBD02, DFGV03, BBL05]. Seuraavassa käydään läpi erilaisia tapoja mallintaa solmujen liiketilaa ja kohtaamisia sekä viestien kulkua verkon läpi.

Jos verkon kaikki solmut ovat liikkeessä, ne kohtaavat ennen pitkää toisensa ja reititystä voidaan kuvata nollan hypyn verkkomallilla eli suoralähetyksellä (engl. direct transmission) [JOW+02, LTZG06, WJMF05]. Siinä lähettäjä odottaa, kunnes vastaanottaja tulee vastaan ja viesti saadaan toimitettua. Ratkaisu on yksinkertainen, mutta käyttökelpoisuudeltaan hyvin rajoittunut.

Realistisempaa on käyttää edes yhtä välittäjäsolmua lähettäjän ja vastaanottajan välillä. Kahden hypyn reitityksessä (engl. 2-hop routing) lähettävä solmu antaa viestinsä välittäjäsolmulle, jonka varastoi sen itsellään kunnes kohtaa vastaanottajasolmun [GT02, GGL07]. Näin kaikkien solmujen ei tarvitse kohdata toisiaan, vaan riittää, että lähettäjäsolmusta on kahden verkkohypyn mittainen polku vastaanottajasolmuun.

Myös kahden hypyn reititys perustuu olettamukseen solmujen liikkuvuudesta eli siitä, että riittävä määrä solmuja tulee toistensa lähelle ennemmin tai myöhemmin. Solmujen mukanaan kuljettamat viestit lisäävät verkon suorituskykyä [GT02]. Tällainen liikkuvuus toteutuu esimerkiksi joukossa konferenssivieraita, joista jotkut liikkuvat riittävän usein eri esityspaikkojen väleillä kuljettaakseen viestejä näiden välillä [DFL01, HCS+05].

Yleiskäyttöisempi reititysmalli on usean hypyn reititys (engl. multihop routing), jossa viesti kulkee useamman välittäjäsolmun kautta [PWWM05, FLZ+05]. Tällöin solmujen ei tarvitse liikkua, vaan riittää, että verkon halki löydetään polku lähettäjältä vastaanottajalle. Kahden hypyn reititys on hyödyllinen alkeistapaus usean hypyn reitityksestä, sillä siinäkin pyritään valitsemaan seuraavaksi välittäjäsolmuksi sellainen solmu, jolla on parhaat mahdollisuudet saada viesti perille.

Merkittävä osa ad hoc -verkkojen reitityksen tutkimuksesta perustuu kahden hypyn reititysmallille. Malli on todellisiin verkkoihin liian yksinkertaistettu, sillä se olettaa solmujen olevan välityskyvyiltään ja liikkeitään samanlaisia. Käytännössä solmujen liikkeiden ja kohtaamisaikojen erot tekevät kahden hypyn reitityksestä joissakin tilanteissa hyvin tehottoman [GGL07, HCS+05].

Liikkuvuuden mallinnus

DTN-verkoissa solmujen liike kuljettaa viestejä verkon solmujen välillä. Solmujen liikkuvuutta voidaan mallintaa joko teoreettisesti tai todellisten käyttötapausten perusteella. Yleisimmät teoreettiset liikkuvuusmallit ovat satunnaisaskel ja satunnaisreittipistemallit [CBD02]. Satunnaisaskelmallissa verkon solmut liikkuvat askelen satunnaiseen suuntaan jokaisella simulaatiokierroksella. Tätä voisi vastata merellä myrskyyn joutunut sensoriverkko.

Hiukan realistisemmassa satunnaisreittipistemallissa solmut valitsevat itselleen uuden satunnaisen sijainnin ja liikkuvat sinne tasaisella satunnaisella nopeudella [JLW05, HABR05, TZZ03, OKD06]. Edelleen todenmukaisempi mallinnus solmujen liikkeelle on yhteisöliikkuvuusmalli (engl. community mobility model), jossa solmut oleskelevat pääasiassa omassa kotiverkossaan, mutta vierailevat tietyllä todennäköisyydellä myös toisissa verkoissa [LDS03, BBL05, WW06]. Viestin reitittäminen toiseen verkkoon onnistuu antamalla se välitettäväksi sellaiselle solmulle, joka on todennäköisesti vierailemassa tuossa toisessa verkossa.

Tällainen vierailija voi olla myös erikseen kyseiseen tehtävään varattu, mahdollisesti tavallista tehokkaampi verkon solmu, joka käy läpi koko verkon, kerää solmujen lähettämät viestit sekä toimittaa saapuvat viestit. Tällaista liikkuvaa solmua kutsutaan tietojuhdaksi (engl. data mule) tai viestilautaksi (engl. message ferry), koska se kuljettaa viestejä verkon solmujen välillä [PFH04, JDPF05, TAZ06]. Esimerkiksi kyliä ja kaupunkeja kiertävä reitityssolmun sisältävä bussi tai sensoriverkon tietoja keräävä keräyssolmu sopivat hyvin mallinnettavaksi viestilauttana.

Solmujen liikkuvuus voi olla seurausta solmujen normaalitoiminnoista, kuten eläinten liikkeistä maastossa tai bussin liikkeestä kaupunkien väleillä. Joissakin tapauksissa liike voi palvella myös itse tiedonvälitystä, jolloin solmut itse liikkuvat aktiivisesti kohtaamaan reittiään kulkevan viestilautan. Jos verkon reititys perustuu tarkkaan maantieteelliseen paikkatietoon, voivat solmut laskea edullisimman reitin sopivimpaan kohtauspaikkaan ja siirtyä sinne [LR03].

Ajoneuvoverkkojen liikkumista voidaan mallintaa laittamalla solmut liikkumaan joko ruu-

tuasemakaavaa tai valtatietä pitkin. Näin solmut kohtaavat toisiaan joko kadunkulmissa tai vastakkaisiin suuntiin menevillä kaistoilla [Ker07]. Valtatiemallissa on otettava huomioon myös samaan suuntaan liikkuvien solmujen erilaiset nopeudet ja näin syntyvät kontaktit.

Kohtaaminen ja viestinvälitys

Kaikissa reititysmalleissa toisensa kohtaavat solmut vaihtavat kuljettamiaan viestejä. Aluksi ne vaihtavat tiivistetyn luettelon viestivarastojensa sisällöstä (engl. summary vector) ja pyytävät sen jälkeen toiselta haluamansa viestit. Kukin solmu on vapaa päättämään, mitä viestejä se ottaa välittääkseen. Solmu voi esimerkiksi suosia tietylle vastaanottajalle meneviä tai tietyn kokoisia viestejä [VB00]. DTN-verkkojen eri reititysalgoritmit pyrkivät löytämään optimiratkaisut vaihdettavien viestien määrille ja valituille välittäjäsolmuille.

Viestin perillemenoa voidaan varmistaa antamalla se useammalle kuin yhdelle välittäjäsolmulle. Lähetettävien viestikopioiden määrällä voidaan hallita reitityksen yleisrasitusta. Kaksihyppyisessä reitityksessä valinnan tekee lähettäjäsolmu, monihyppyisessä reitityksessä myös muut solmut voivat edelleen monistaa viestiä omien reititystietojensa perusteella.

Solmut voivat vaihtaa viestien lisäksi myös reititystietoja eli tietoa siitä, mitä muita solmuja ja palveluita verkossa on saavutettavissa. Tämä lisää hallinnollista liikennettä, mutta tuottaa useampihyppyisen, transitiivisen kuvan verkosta [DFGV03, TZZ03, LTZG06, WW06].

Reititystä voidaan tehostaa sitä enemmän, mitä tarkempaa tietoa verkon kontakteista on käytettävissä [TZZ03, WW06, JDPF05, MLS06, BGJL06]. Ulkoisten lähteiden lisäksi tätä tietoa voidaan hankkia seuraamalla, miten verkon solmut kohtaavat toisiaan ja tehdä tästä kohtaamishistoriasta ennusteita tuleville kontakteille. Ennusteiden tarkkuuteen vaikuttavat huomattavasti solmujen todellisten liikkeiden säännönmukaisuudet tai niiden puute. Satunnaisesti liikkuvien solmujen välisiä kontakteja ei voida ennustaa.

Näissä kohtaamismalleissa oletetaan, että solmujen kohtaamishistoriasta voidaan suoraan ennustaa tulevaisuutta, eli että solmut kohtaavat toisiaan tulevaisuudessa samalla tavalla kuin ennenkin. Esimerkiksi ajoneuvoverkoissa, joissa solmut kohtaavat toisensa vain harvoin, tämä olettamus ei selvästikään toteudu [LTZG06, BGJL06].

Solmujen kohtaamistiheyttä τ arvioidaan yleensä yksinkertaisesti laskemalla miten usein ne ovat keskimäärin kohdanneet tiettynä aikana $\tau = \frac{kohtaamiset}{aikaikkuna}$ [DFGV03, CYS+06, BBL05, JLW05, LTZG06, WW06, WW07]. Kukin solmu ylläpitää taulukkoa, jossa on arvio verkon muiden solmujen kohtaamistiheydestä. Lisäksi voidaan ylläpitää tietoa viimeisen kohtaamisen ajanhetkestä, yhteyden kestosta ja kohtaamisen fyysisestä sijaintipai-

kasta sekä muistitilan käytöstä. Toimimalla viimeisestä kohtaamishetkestä kuluneen ajan perusteella, ei reitityksessä tarvita koko verkon kattavaa aikasynkronointia [DFGV03].

Verkon solmujen viestinvälityskykyä mallinnetaan toimitustodennäköisyydellä (engl. delivery predictability). Solmujen toimitustodennäköisyystaulukko kertoo, millä todennäköisyydellä kohdattu solmu voi toimittaa viestin perille muihin solmuihin. Todennäköisyys on aluksi nolla, mutta se kasvaa solmujen kohdatessa toisiaan [LDS03]. Kohtaamistietoja pitää vanhentaa, jotta viestejä ei reititetä vanhojen yhteyksien perusteella.

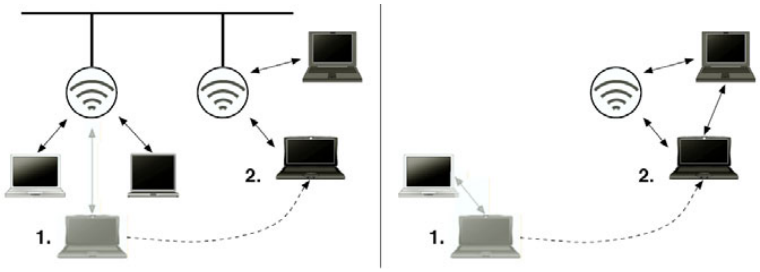
Yksinkertaisin malli tietyn solmun käyttökelpoisuudelle viestien välittäjänä on suoraan sen havaittu kohtaamistiheys. Todellisuudessa olisi parempi käyttää aritmeettisen keskiarvon sijaan eksponentiaalisesti painotettua keskiarvoa, Markovin ketjuihin perustuvaa mallia [BBL05] tai solmujen omaa ilmoitusta halukkuudestaan kuljettaa viestejä [HABR05].

Toimitustodennäköisyys on haluttaessa transitiivinen ominaisuus, jolloin solmut voivat oppia verkon kaikkien solmujen välityskykyjä [TZZ03]. Reittien virkistämisessä, vanhentamisessa ja transitiivisuudessa voidaan käyttää erilaisia painoarvoja ja algoritmeja solmujen liiketilamallin mukaan [LDS03, WW06, BGJL06].

Kohtaamistapahtumien ennustamisen lisäksi voidaan mallintaa solmujen vierailua tietyssä paikassa. Tällöin voidaan esimerkiksi sijoittaa eläimiin kiinnitettyjen sensoreiden tiedonkeräysasema sopivaan ruokintapaikkaan [BBL05, JOW+02]. Paikkana voidaan käyttää tietyn tukiaseman peittoaluetta tai todellistakin maantieteellistä sijaintia, jos sellainen on saatavilla esimerkiksi GPS-satelliittijärjestelmän kautta. Paikkapohjaisessa reitityksessä viestejä voidaan lähettää johonkin tiettyyn paikkaan, jolloin kaikki sen läheisyydessä olevat solmut vastaanottavat viestin [SK07]. Havaintojen mukaan yleisimmin käytetyt algoritmit tarvitsevat tuhansia havaintoja päästäkseen 70 % tarkkuuteen solmun seuraavan sijaintipaikan ennustamisessa [SKJH03].

Edellä esitettyjen solmujen todelliseen liikkumiseen perustuvien mallien lisäksi liikkuvuus voi olla myös virtuaalista. Tällöin ajatellaan, että solmut ovat koko ajan toistensa lähetysalueella tai voivat muuten vaikuttaa siihen, milloin yhteyksiä voidaan muodostaa. Näitä pyynnöstä muodostettavia kontakteja voidaan mallittaa samalla tavalla, kuin solmujen välistä todellista liikkuvuutta. Todellisesta tai virtuaalisesta liikkuvuudesta voidaan muodostaa useampiulotteinen liikkuvuusavaruus, jossa solmuille voidaan määritellä euklidiset etäisyydet ja niiden muutokset [LFC05]. Solmujen välisiä kontakteja voidaan etsiä tällaisesti virtuaalisesta avaruudesta hyvin abstraktilla tasolla.

Kaikki historiaan perustuvat ennustusmallit tarvitsevat aluksi lämmittelyajan (engl. warm-up period), jonka kuluessa kerätään tietoja solmujen välisistä kohtaamisista. Lämmittely-



Kuva 11: Langattoman verkon asiakaskoneiden liikkuvuustietoja voidaan käyttää löytämään solmuja, kuten harmaalla merkitty kannettava tietokone, jotka voisivat kuljettaa viestejä verkkojen välillä [JLW05].

aikana viestit pitää välittää tulvittamalla, koska käytettävissä ei ole parempaakaan tietoa eri solmujen sijainnista tai niiden sopivuudesta viestien välittäjiksi.

Todelliset liikkuvuusaineistot

Verkkoprotokollien suorituskykyä tutkitaan yleensä verkkosimulaattoreilla, sillä varsinaisia DTN-testiverkkoja ei ole olemassa [LGE06]. Teoreettisten skenaarioiden lisäksi verkon toimintaa voidaan tutkia todellisista verkoista kerättyjen yhteyshistorioiden avulla. Tällöin käytetään esimerkiksi kuvan 11 esittämällä tavalla jonkin langattoman lähiverkon tukiasemien piirissä olleiden laitteiden sijaintihavaintoja ja muutetaan ne tutkittavan verkon liikkuvuustiedoiksi [SKJH03, JLW05, SK07, KKK06]. Samoin voidaan muodostaa myös Bluetooth-solmujen ryhmääntymistä kuvaavia aineistoja [McD05, LLS+06]. Niiden avulla voidaan löytää solmuja, joita voidaan käyttää viestien siirtämiseen eri verkkojen väleillä [JLW05].

Muita vastaavia aineistoja ovat esimerkiksi bussien kulku reitillään, eläinten liikkeet maastossa tai konferenssivieraiden liikkeet eri esityssaleissa. Näitä liikkuvuustietoja on saatavilla tutkijoiden käyttöön CRAWDAD-palvelussa [HCS+05, BGJL06, SKZ06, ZKL+07, KHA07, LLS+06]. Solmujen tuottaman verkkoliikenteen tunnuslukuja voidaan mallittaa samoista lähteistä tai yleisistä Internetin tilastoista [KRH+06, SPH05]. Näitä realistisia liikkuvuusaineistoja voidaan käyttää myös kontaktien ennustusmenetelmien testaamiseen tutkimalla miten hyvin ennustettu ja toteutunut kontakti vastaavat toisiaan.

Talletustilan hallinta

DTN-verkkojen toiminta perustuu välitettävien viestien tallettamiseen verkon solmuihin (engl. in-network storage). Solmun on hyvä rajoittaa muiden solmujen viesteille varaamaansa talletustilaa sekä valita, millä ehdoilla se ottaa uusia viestejä säilytettäväkseen [VB00]. Solmun resurssien loputtua sen on poistettava viestejä lähetysjonosta.

Poistettavien viestien valinta eli pudotusstrategia (engl. drop strategy) vaikuttaa välitettävien viestien toimitusviiveeseen ja -varmuuteen [TZZ03]. Simulaatioiden mukaan tehokkainta on pudottaa pisimpään verkossa ollut viesti (engl. drop oldest) tai sellaisen solmun viesti, jonka kohtaamisesta on kulunut pisin aika (engl. drop least encountered) [DFL01]. Jos reitityksessä ei pidetä yllä arviota toisten solmujen kohtaamisista, on tehokkainta pudottaa joko vanhin samalta välityssolmulta saatu viesti tai pisimpään säilytetty viesti [ZNKT07]. Palvelunlaatuluokkia käytettäessä pudotetaan tasatilanteessa matalimman prioriteetin tai vähiten vikasietoinen viesti [WW07, RHB+07].

Yhteyskerroksessa käytettävä aktiivinen [BGJL06] tai passiivinen [HABR05] kuittausmenettely on tehokas ja eksplisiittinen tapa saada tietoon vanhentuneiden viestien tunnisteet. Niiden käyttäminen kuitenkin lisää verkon kuormitusta, koska jokainen viesti pitää kuitata verkossa vastakkaiseen suuntaan kulkevalla sanomalla.

Mikäli mitään tietoa solmujen reitityskyvyistä ei ole käytettävissä, ei ole merkitystä, mille solmulle viesti annetaan. Tällöin viestin reititys muuttuu satunnaiseksi (rajoitettu määrä kopioita) [BGJL06, SK07] ja ääritapauksessa epideemiseksi (rajoittamaton määrä kopioita) [VB00].

Edellä on käyty läpi erilaisten reititysmallien kuvaamiseen tarvittavaa käsitteistöä. Seuraavassa tarkastellaan tärkeimpiä reitityksen ja viestinvälittämisen ratkaisuja yksinkertaisimmasta eli epideemisestä reitityksestä alkaen.

3.4 Epideeminen reititys

Kaikkein yksinkertaisin verkkojen viestinvälitystapa on tulvitus (engl. flooding), jossa jokainen solmun vastaanottama viesti kopioidaan edelleen jokaiselle naapurisolmulle. Näin viesti leviää verkkoon solmuväli kerrallaan ja saavuttaa vastaanottajan ennen pitkää. Tulvituksen toteuttamisessa ei tarvita minkäänlaista muistia lähettäjistä tai vastaanottajista, mutta syntyvän valtavan liikennemäärän vuoksi se on käyttökelpoinen vain pienissä verkoissa.

Epideeminen reititys toimii samalla tavalla, mutta siinä viestiä ei kopioida siihen suuntaan, josta se vastaanotettiin. Jokainen solmu pitää kirjaa, keneltä se on minkäkin viestin saanut eikä samaa viesti välitetä kuin kerran. Vastaanottajasolmu saa viestistä kopion jokaiselta naapuriltaan. Epideemisessä reitityksessä viesti leviää verkossa kuin tarttuva sairaus ja “tartuttaa” lopulta verkon jokaisen solmun [DGH+88, VB00].

Epideeminen reititys on teoreettisesti nopein tapa saada viesti perille, sillä jokaisen solmun kautta kulkeva viesti löytää automaattisesti myös nopeimman polun lähettäjältä vas-

taanottajalle. Tämän suorituskyvyn hintana on verkon kasvaessa hyvin nopeasti kasvava resurssien kulutus [LDS03, SPR05].

Epideeminen reititys saattaa olla kriittisissä tapauksissa ainoa ratkaisu, sillä erittäin huonoissa verkko-olosuhteissa kehittyneemmät protokollat eivät toimi luotettavasti. Ruuhkaantuneessa verkossa epideeminen reititys toisaalta tuhlaa jo muutenkin niukkoja resursseja ja pahentaa ruuhkaa entisestään.

Yksinkertaisin tapa vähentää epideemisen reitityksen kuormaa on vähentää viestistä syntyvien kopioiden määrää jollakin algoritmilla [ZNKT07, LDS03]. Erilaisissa verkoissa ja sovelluksissa tarvitaan erilaisia ratkaisuja.

Kopioiden vähentäminen

Viestin aikaleimakentän perusteella voidaan viestin leviäminen pysäyttää tietyn ajanhetken tai verkkohyppymäärän jälkeen [HABR05]. Toinen yksinkertainen ratkaisu vähentää viestistä tehtävien kopioiden määrää on lähettää kopioita vain osassa, esimerkiksi kymmenessä ensimmäisessä tai satunnaisesti arvotussa kontaktissa.

Tiiviissä verkossa solmujen välisiä yhteyksiä on niin paljon, että pienikin määrä viestikopioita pääsee perille nopeasti. Näissä verkoissa kopioita voidaan vähentää ilman, että viestin toimitusviive juurikaan kasvaa [JDPF05, WW06]. Sen sijaan harvassa verkossa suuri osa kopioista päätyy odottamaan yhteyksiä tai hukkuu kokonaan, joten kopioiden vähentäminen kasvattaa toimitusviivettä hyvin nopeasti [VB00, WJMF05].

Osa täydelliseen verkon tilannekuvaan pyrkivistä reititysalgoritmeista lähettää verkkoon vain yhden viestikopion. Tämä toimintatapa ei ole kovin vikasietoinen, sillä tämä ainoa viesti voi hukkua matkalla [JLW05, SPR04a]. Kopioiden määrää voidaan käyttää myös tukemaan liikenteen palvelunlaatuluokkia, tekemällä esimerkiksi kiireellisistä eli tärkeistä viesteistä enemmin kopioita kuin muiden luokkien viesteistä [WW07, SK07].

Probabilistinen reititys

Epideeminen reititys olettaa, että verkon solmut ovat viestinvälityskyvyltään samanlaisia ja liikkuvat satunnaisesti. Todellisissa verkoissa nämä oletukset eivät ole voimassa, vaan solmujen liikkeissä esiintyy jaksollisuutta ja ryhmäytymistä. Probabilistinen reititys hyödyntää näitä säännönmukaisuuksia valitessaan viestikopioiden välittäjäsolmuja [LDS03]. Näin viestinvälityksen tehokkuus paranee ja verkkoon lähetettävien viestikopioiden määrä vähenee.

Epideemisessä reitityksessä kohtaavat solmut synkronoivat täysin viestipuskuriensa sisällöt. Probabilistisessa reitityksessä solmut lähettävät toisilleen vain ne viestit, joiden vastaanottajien toimittajana toinen solmu olisi niitä itseä parempi. Samalla solmut voivat

valvoa tehtävien viestikopioiden määrää ja esimerkiksi lakata levittämästä viestiä, josta ne ovat antaneet kopion riittävälle määrälle riittävän hyviä välittäjäsolmuja.

Yhteisöpohjaisessa liikkuvuusmallissa probabilistinen reititys tuottaa pienemmän viiveen ja vähemmän viestikopioita kuin epideeminen reititys [LDS03]. Sama ero näkyy myös satunnaisessa liikkuvuusmallissa, koska siinä solmut kohtaavat lähiaikoina todennäköisimmin niitä solmuja, joiden lähellä ne ovat hiljattain olleet. Probabilistinen reititys pystyy hyödyntämään näitä pieniäkin poikkeamia puhtaasta satunnaisuudesta.

Viestien suuntaaminen

Suunnatussa tulvituksessa (engl. directed flooding) levitetään viestin kopioita vain sellaisten solmujen kautta, jotka ovat samalla suunnalla kuin vastaanottaja. Tässä suunta tarkoittaa viestin reitityspolun määrittelemää reittiä verkon läpi, ei todellista geometrista suuntaa. Tällöin viestiä ei turhaan levitetä sellaisiin osiin verkkoja, jossa vastaanottaja ei viestin elinaikana tule todennäköisesti käymään [SPR04a, SPR04b, SPR05, HABR05].

Puolittava suunnattu lähetys (engl. binary spray and wait) toimii kahdessa vaiheessa. Levitysvaiheessa (engl. spray) lähettäjä antaa puolet viestikopioistaan kohtaamalleen solmulle. Solmu toimii tällä tavalla, kunnes jäljellä on vain yksi kopio. Tämän jälkeen solmu siirtyy odotusvaiheeseen eli odottamaan vastaanottajan kohtaamista suoralähetystä varten. Jokainen viestikopion saanut välittäjäsolmu toimii vastaavalla tavalla. Syntyvien kopioiden määrä voidaan valita haluttavan toimitusviiveen mukaan, sillä puolittamisen seurauksena protokollan yleisrasitus riippuu vain verkon solmujen määrästä, ei itse verkon koosta [SPR05].

Suunnattu lähetys ei ole kuitenkaan kovin tehokas, sillä toimiessaan, kuten aikarajoitettu epideeminen reititys, se ei yritä hyödyntää solmujen erilaisia liikkuvuustapoja. Siitä edelleen kehitetty tarkentava suunnattu lähetys (engl. spray and focus) voi hyödyntää jo odotusvaiheessa tarkempaa vastaanottajan saavutettavuustietoa [SPR04b].

Yhteyksien ennustaminen

Vaikka varsinaisen viestiliikenteen välittämiseen epideeminen reititys saattaa olla liian raskas, voidaan sitä käyttää välittämään reititystietoa solmujen välillä. Näin solmut saavat selville verkon nykyiset ja ennustetut kontaktit ja pystyvät reitittämään viestejä joko lähdereitityksellä tai askel kerrallaan. Tämä MEED-reititysprotokolla (minimum estimated expected delay) käyttää havaittuja kontakteja luomaan ennusteen eri solmujen välisistä yhteyksistä ja viestinvälityskyvyistä [JLW05].

Verkon yhteystietojen levittäminen kuluttaa solmujen resursseja, joten jokaista pientä muutosta ei kannata toimittaa koko verkkoon. Simulaatioiden mukaan 5 % hystereesi

reitityskyvyn muutoksien levittämisessä verkon muille solmuille vähentää topologiapäivitysten yleisrasitusta riittävästi [JLW05].

Tällainen täydelliseen verkon topologiatietoon perustuva reititysprotokolla antaa mahdollisuuden käyttää vain yhtä viestikopiota. Tällöin otetaan riski tämän ainoan kopion katoamisesta epäluotettavan solmun tai verkkoon syntyvien silmukoiden vuoksi. Tätä riskiä voidaan pienentää käyttämällä viestien välittämiseen esimerkiksi seuraavaksi esiteltäviä koodaustekniikoita [CYS+06].

3.5 Viestien koodaus

Informaatioteorian mukaan viesti voidaan toimittaa perille joissakin tapauksissa tehokkaammin lähettämällä viestin sijasta verkkoon joukko siitä tuotettuja koodilohkoja, "todisteita" [Sha48, EKM07]. Vastaanottaja voi rekonstruoida alkuperäisen viestin kerättyään verkosta riittävän määrän näitä todisteita [WJMF05, LTZG06, CYS+06, WB05, FBW06, KRH+06, EKM07, LLL07]. Voidaan myös osoittaa, että epäluotettavilla yhteyksillä riittävä määrä todisteita pääsee vastaanottajalle nopeammin, kuin varsinainen viesti olisi päässyt. Tämä nopeutus syntyy siitä, että pienet koodilohkot kulkevat verkossa nopeammin, kuin suuri alkuperäinen viesti olisi kulkenut, sekä siitä, että minkään yksittäisen koodilohkon katoaminen ei estä viestin toimittamista.

Perinteisessä reitityksessä voidaan hyödyntää vain nopeimpia ja luotettavimpia yhteyksiä. Viestin pienenkin osan lähettäminen epäluotettavan yhteyden kautta riskeeraa koko viestin perillemenon, sillä kaikkien osien on päästävä vastaanottajalle. Koodausratkaisuissa mikään viestin osa ei ole korvaamaton, vaan riittää, että riittävä määrä erilaisia koodilohkoja pääsee perille. Koodaus leikkaa siten viestien toimitusaikajakaumasta pitkän hännän pois [WJMF05, LTZG06, CYS+06]. Viesti voidaan antaa sovellukselle heti, kun riittävä määrä sen osia on saatu perille, eivätkä vielä matkalla olevat osat kasvata toimitusviivettä.

Tarvittavien viestilohkojen määrää ei tiedetä täsmällisesti ennakolta, joten haluttuun toimitusvarmuuteen sisältyy tietyn suuruinen riski. Koodauksen pettäessä hukkuvasta viestistä toivutaan tavalliseen tapaan uudelleenlähetyksellä. Koodausratkaisuissa tiettyä yleisrasitetta vastaa kuitenkin huomattavasti vastaavaa viestien epideemistä kopiointia parempi toimitusvarmuus [CYS+06]. Koodauksen hyödyt tulevat esille jo kohtuullisillakin välittäjäsolmujen määrillä. Perusmuodossaan jo 32 välittäjällä eli lohkolla viiveen keskiarvo on enää 10 % päässä äärettömän monen välittäjän tuottamasta viiveen keskiarvosta [WJMF05]. Pienelläkin määrällä koodausta päästään siten hyvään suorituskykyyn.

Perinteisesti koodausratkaisuja on käytetty havaitsemaan ja korjaamaan siirtovirheitä, mut-

ta tällä tavalla käytettynä ne auttavat hallitsemaan DTN-verkkojen viivettä etenkin epäluotettavilla yhteyksillä [WJMF05, WW06]. Koodaus tuo ylimääräistä viivettä silloin, kun verkkoyhteys toimii moitteetta eli toimitusviive on pieni. Koodauksen tuoma lisäviive ei ole merkittävä silloin, kun itse verkosta syntyvä viive on esimerkiksi huonojen yhteyksien vuoksi suuri. Optimaalinen reititysratkaisu valitsee kulloiseenkin verkkoympäristöön sopivan toimintatavan eli vähentää koodauksen käyttöä hyvin toimivassa verkossa. Siirtokerroksen toiminta tapahtuu reitityksen alla ja on sen suhteen ortogonaalinen, joten koodausta voidaan käyttää myös kaikkien muiden reititysratkaisujen kanssa samaan aikaan [WJMF05, CYS+06, LTZG06].

DTN-verkoissa sovellettaviksi eniten tutkittuja koodauksia ovat poistokoodaus (engl. erasure coding) [WJMF05] ja verkkokoodaus (engl. network coding) [WB05]. Poistokoodauksessa viestin lähettäjä koodaa ja vastaanottaja purkaa, verkkokoodauksessa näitä kumpaakin tehdään jokaisella solmuvälillä erikseen.

Poistokoodaus

Poistokoodauksessa (engl. erasure coding)[1] viesti muutetaan lähetyssolmussa riippumattomiksi koodilohkoiksi ja kootaan vastaanottajasolmussa [WJMF05]. Poistokoodaus on muunnelma ennakoivasta virheenkorjaavasta koodista (engl. Forward Error Correction, FEC), jossa lähettäjä lisää viestiin redundanttia informaatiota, jonka avulla vastaanottaja pystyy rekonstruoimaan matkalla vääristyneen viestin.

Koodattaessa kooltaan M oleva viesti valitulla toistokertoimella r saadaan aikaan $\frac{M \times r}{b}$ kappaletta kooltaan b olevaa viestilohkoa siten, että mitkä tahansa $(1+\epsilon) \times \frac{M}{b}$ koodilohkoa riittävät viestin dekoodaamiseen. Tässä ϵ on koodausalgoritmikohtainen pieni vakio, joka kertoo koodauksen yleisrasitteen. Yleisimmillä koodausratkaisuilla ϵ on luokkaa 5-10 % [CWJ03, Mit04, WB05].

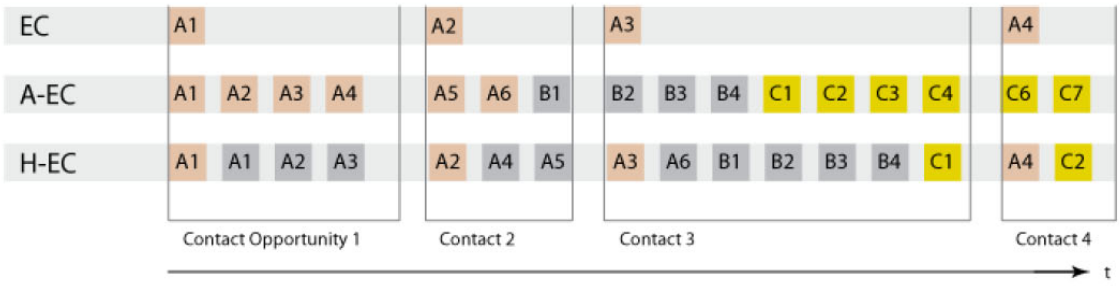
Koodausratkaisujen hyöty on siinä, että kertoimella r koodatun viestin rekonstruoimiseen riittää vastaanottaa mitkä tahansa $\frac{1}{r}$ koodilohkoa [WJMF05, CYS+06]. Poistokoodausta on tutkittu pääasiassa kahden hypyn reititysmallilla, jossa tuotetut koodilohkot välitetään $k \times r$ välityssolmulle. Tässä k on haluttu levitysvakio. Pelkään kopiointiin verrattuna tässä käytetään k-kertainen määrä välittäjiä, mutta jokaiselle välittäjälle tulee vain $\frac{1}{k}$ liikenteestä. Kaikkiaan tavuja tuotetaan $r \times M$, mikä on sama kuin lähettämällä r kappaletta alkuperäistä kooltaan M olevan viestin kopioita. Koodauksen käyttö ei siten käytännössä lisää liikenteen määrää. Jos $k = 1$, koodaus toimii kuten kopiointi, eli alkuperäinen viesti annetaan sellaisenaan r:lle välittäjälle.

[1] Poistokoodaus on saanut nimensä siitä, että sen avulla toivutaan bittejä hukkaavan siirtotien (engl. erasure channel) tuottamista virheistä.

Eri välittäjäsolmuille annetut koodilohkot etenevät verkossa eri reittejä, joista jotkin ovat keskimääräistä nopeampia. Jos näitä nopeampia reittejä on vähintään k kappaletta, saadaan koodaamalla viesti perille nopeammin kuin kopioimalla. Kyse onkin siitä, onko parempi käyttää r:ää välityssolmua ja odottaa yhden onnistuvan viestin toimituksessa vai $k \times r$:ää ja odottaa k:n onnistuvan.

Valitsemalla koodausparametrit k, r ja b halutuiksi, päästään suurella todennäköisyydellä vakiomittaiseen viiveeseen [WJMF05]. Koodauksen parametrisointia voidaan käyttää myös eri liikenneluokkien toteuttamiseen niin, että tärkeästä liikenteestä tuotetaan enemmän koodilohkoja kuin perusliikenteestä. Tärkeä liikenne voidaan myös lähettää verkkoon ennen muuta, jolloin ne myös vastaanotetaan todennäköisesti aikaisemmin kuin muut viestit [LLL07]. Tämän tehokkuus tosin laskee jyrkästi, jos kiertoviive on suuri.

Erittäin huonoissa olosuhteissa voidaan viestikopioiden määrää r lisätä vielä suuremmaksi (engl. adaptive coding and forwarding). Vastaavasti hyvissä oloissa sitä voidaan pienentää. Verkon olosuhteiden selvittäminen vaatii joko aktiivista mittausta tai verkonvalvonnan keräämää tietoa, joka toimitetaan solmuille erillisillä ohjausviesteillä.



Kuva 12: Poistokoodaus EC lähettää viestin A koodilohkot kunkin omassa kontaktissaan. Tehokkaampi A-EC käyttää jokaisen kontaktin kokonaan, mikä laskee vikasietoisuutta. Hybridimalli H-EC lähettää kaksi kopiota kustakin lohkosta, mutta eri kontaktien kautta [CYS+06]

.

Kuva 12 esittää poistokoodauksen eri varianttien toimintaa. Perusmuotoinen poistokoodaus (EC) lähettää vain yhden lohkon jokaiselle kontaktille, jolloin jokaisesta kontaktista saattaa jäädä lähetysaikaa käyttämättä [WJMF05]. Tätä voidaan tehostaa lähettämällä aggressiivisesti (A-EC) jokaisessa kontaktissa niin monta lohkoa, kuin sen kesto sallii. Näin jokaisen kontaktin koko yhteysaika saadaan käyttöön. Ongelmaksi tulee tällöin viestin kaikkien lohkojen lähettäminen huonosti toimivalle solmulle, joka ei koskaan saa toimitettua niitä perille. Realistisissa käyttötapauksissa tällaisten musta aukko -solmujen

olemassaoloon pitää varautua [CYS+06].

Yhdistelmäratkaisu H-EC lähettää koodilohkot verkkoon kahtena kopiona niin, että ensimmäisen kopion lohkot lähetetään kukin omalla kontaktillaan ja toisen kopion lohkoilla täytetään muiden kontaktien lähetysaika. Näin jokaisesta viestistä lähetetään suorituskyvyn parantamiseksi kaksi kopiota, mutta vikasietoisuus turvataan koodauksella. Näin musta aukko -solmujen olemassaolo verkossa ei estä viestien perillemenoa. Hybridiratkaisu yhdistää koodauksen vikasietoisuuden ja replikoinnin suorituskyvyn, eikä hukkaa resursseja, koska jokaisen kontaktin aikana lähetetään maksimimäärä viestilohkoja. Kahden viestikopion tekeminen on hyväksyttävää useimmissa käyttötapauksissa [CYS+06].

Historiapohjainen poistokoodaus

Yhdistämällä koodaus historiapohjaiseen reititykseen voidaan edelleen parantaa reitityksen tehokkuutta [LTZG06]. Tekniikka perustuu kontaktien ennustamiseen tai jonkin muun solmujen välityskykyä kuvaavan parametrin käyttöön välittäjien valinnassa.

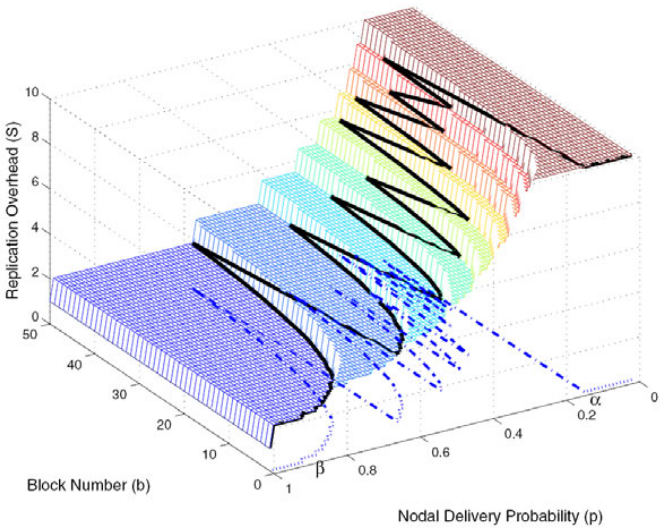
Historiapohjainen poistokoodaus toimii kuten luvussa 3.4 sivulla 35 esitelty puolittava suunnattu lähetys (engl. binary spray and wait) [SPR05]. Siinä solmu siirtyy levitystilasta odotustilaan, kun sillä on vähemmän kuin ennalta asetettu määrä G viestilohkoa välitettävänään. Aluksi solmu on levitystilassa ja sillä on kaikki $r \times k$ lohkoa, missä r on viestikopioiden määrä ja k tehtävien koodilohkojen määrä. Viestin koostamiseksi riittää, että vastaanottaja saa mitkä tahansa k lohkoa. Jos $k = 1$, välitetään jokainen viesti yhtenä kokonaisuutena parhaan välittäjäsolmun kautta. Tällöin on mahdollista, että valittu solmu ei kuitenkaan pysty välittämään viestiä perille. Jos taas $r = 1$, ei välittäjiä käytetä, vaan ainoa viestikopio annetaan vain vastaanottajalle.

Jokaisessa kontaktissa solmu antaa pois kohtaamansa solmun viestinvälityskykyyn suhteutetun osuuden viestikopioistaan. Kun solmu on antanut pois G lohkoa, se siirtyy välitystilaan, jossa se joko antaa kaikki lohkonsa kohdatulle solmulle tai ottaa solmulta kaikki sen lohkot, sen mukaan kummalla niistä on parempi välityskyky.

Lohkot leviävät aluksi tehokkaasti useille poluille ja sen jälkeen niitä annetaan vain entistä paremmille välityssolmuille [LTZG06, WW06]. Osa lohkoista voidaan antaa myös huonommin toimiville poluille, koska koodaus osaa hyödyntää myös keskimääräistä hitaammin kulkevia lohkoja. Tällä tavalla voidaan ottaa huomioon myös solmujen erilaiset mahdollisuudet välittää liikennettä.

Historiapohjainen poistokoodaus toimii parhaiten silloin, kun verkkoyhteydet ovat suhteellisen luotettavia ja yhteydet ennustettavia. Perusmallinen poistokoodaus toimii parem-

min, jos yhteydet katkeilevat, viiveet ovat vaikeasti ennustettavia ja viiveen minimointi on tärkeää [WJMF05, LTZG06].



Kuva 13: Poistokoodauksen toimitustodennäköisyyden (p) vaihtelu kopiomäärän (S) ja lohkojen lukumäärän (b) funktiona [WW06].

Vaihdettavan lohko-osuuden lisäksi solmujen välityskykyä voidaan käyttää myös valitsemaan poistokoodauksen parametreja r ja k. Näin voidaan valita sovellukselle sopiva toimitusvarmuus ja sallittu yleisrasite [WW06]. Simulaatioiden perusteella hyvin pienillä ja suurilla välityskyvyn arvoilla viesti kannattaa kopioida sellaisenaan ja koodata muuten. Pienillä arvoilla koodaus taas tuottaa liikaa yleisrasitetta. Näiden ääriarvojen välillä tarvittavien lohkojen määrä vaihtelee kuvan 13 paksun käyrän mukaan ei-monotonisesti välityskyvyn muuttuessa. Vaihtelun syy on yleisrasituksen kasvun ja perillemenon paranemisen epälineaarinen vuorovaikutus [WW06].

Suihkulähdekoodaus

Poistokoodauksen lisäksi viivesietoisiin verkkoihin on ehdotettu suihkulähdekoodeja[2] (engl. digital fountain codes, LT codes, Raptor codes). Perusmuotoiset koodaukset toimivat tietyllä kiinteällä symbolinopeudella, mikä tekee niistä ongelmallisia muuttuvissa verkko-olosuhteissa sovellettaviksi. Toinen ongelma vakionopeuksisissa koodeissa on niiden suorituskyvyn huononeminen viestikoon kasvaessa, koska pieni pakettikoko edellyttää suurta lohkojen määrää [VSFA07].

Suihkulähdekoodit generoivat äärellisestä lähtöjoukosta potentiaalisesti äärettömän mon-

[2]Samaan tapaan, kuin astian täyttämisessä suihkulähteestä ei olla kiinnostuneita, mitkä pisarat astian täyttävät, saadaan viesti perille keräämällä jatkuvasta koodivirrasta riittävä määrä lohkoja.

ta koodilohkoa. Vastaanottaja voi koostaa lähetetyn viestin, kunhan se on kerännyt koodilohkoja vain muutaman prosentin enemmin, kuin itse viesti on kooltaan [Mit04]. Koodaus ja dekoodaus ovat laskennallisesti yksinkertaisia, eivätkä tuota liiallista yleiskuormaa viestien siirtoon.

Suihkulähdekoodit soveltuvat erityisen hyvin luotettavan ryhmälähetyksen toteuttamiseen. Riittää, että lähettäjä lähettää viestin koodaavia viestilohkoja tietyn ennalta sovitun ajan ja vastaanottajat ovat verkossa niin pitkään, että saavat viestin purettua. Satunnaiseen aikaan verkkoon tuleva solmu voi vastaanottaa alusta alkaen hyödyllisiä (engl. innovative), eli edellisistä lohkoista lineaarisesti riippumattomia, viestilohkoja.

Suihkulähdekoodien käyttöönottoa hidastavat koetun monimutkaisuuden sekä sopivien käyttötapauksien puutteen lisäksi alalle harvinaiset patenttikiistat. Tärkeimmät koodien sovellukset sekä algoritmit ovat yksityisen Digital Fountain -yrityksen lisenssin alaisia, vaikka sen käyttämästä Raptor-koodauksesta onkin valmistumassa avoin Internet-standardi [LSS07].

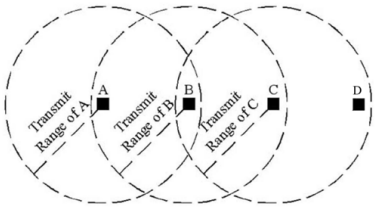
Jos viestien ajatellaan kulkevan verkossa useamman solmun kautta, eikä verkkoa mallineta enää kahden hypyn periaatteella, on ehdotettu esimerkiksi energiatehokkuuden parantamiseksi koodauksen tekemistä myös verkon välittäjäsolmuissa [VSFA07]. Tällöin viestiä koodataan lähettäjän lisäksi myös välittäjäsolmuissa seuraavaksi esiteltävän verkkokoodauksen tapaan.

Verkkokoodaus

Lähettäjäpohjaisissa koodauksissa, kuten poisto- ja suihkulähdekoodauksessa, viestin koodaa vain lähettäjä ja koostaa vastaanottaja. Koodauksen ajatellaan tapahtuvan sovellustasolla eli verkon välittäjäsolmuilla ei ole siinä aktiivista roolia. Verkkokoodaus (engl. network coding) tapahtuu nimensä mukaisesti verkossa, käytännössä siirtokerroksessa. Reitityksen sijaan verkon solmut yhdistävät eri suuntiin meneviä viestejä uusiksi viesteiksi, jotka linkin toisessa päässä olevat solmut purkavat ja mahdollisesti koodaavat uudelleen.

Verkkokoodaus on varsin uusi informaatioteorian sovellutusalue [Sha48, FBW06, EKM07]. Perinteiset tiedonsiirtoväylät on rakennettu pitämään samaan suuntaan menevät viestit irrallaan toisistaan. Paketit ja bitit liikkuvat verkossa samaan tapaan kuin autot valtateillä eli peräkkäin ja risteyskohdissa vain kulkusuuntaansa muuttaen. Verkkokoodaus perustuu eri suunnista tulevien verkkopakettien matemaattiseen yhdistämiseen ennen niiden lähettämistä eteenpäin. Yhdistäminen tehdään tavalla, joka voidaan vastaanottajasolmussa purkaa. Koska yhdistäminen voidaan tehdä hukkaamatta informaatiota, saadaan tällä tavalla lähetettyä yhdessä paketissa tietoa useamman paketin sisällöstä [FBW06].

Taulukko 2: Solmut A ja C lähettävät viestit a ja c toisilleen välittäjäsolmun B kautta. B saa toimitettua molemmat viestit yhdellä lähetyksellä, sillä A ja C ovat sen kuuluvuusalueella ja esimerkiksi A saa viestin c selville laskemalla $c = a \oplus (a \oplus c)$ [FBW06].



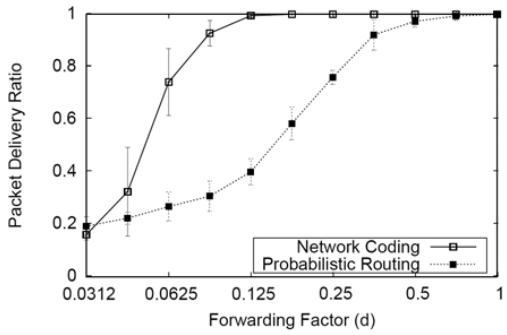
aika ↓					
	A	$\xrightarrow{a}$	B		C
	A		B	$\xleftarrow{c}$	C
	A	$\xleftarrow{a \oplus c}$	B	$\xrightarrow{a \oplus c}$	C

Koodauksen ulottaminen verkon jokaiseen solmuun muuttaa verkon toimintaa merkittävästi. Siinä missä perinteiset reitittimet vain lähettävät paketteja eteenpäin ja ruuhkautuvat, kun yhteyden maksimikapasiteetti saavutetaan, voi verkkokoodain lähettää samalle yhteydelle sen teoreettista suorituskykyä vastaavan määrän informaatiota. Koodaavan verkon yhteydet pystyvät toimimaan tehokkaammin ja vikasietoisemmin, sillä samaa viestiä koodaavia riippumattomia lohkoja voidaan lähettää useampina kopioina rinnakkain pitkin useampia yhteyksiä saamatta aikaan verkkosilmukkaa [EKM07].

Verkkokoodausta voidaan käyttää sekä kiinteissä että langattomissa verkoissa, kunhan tarjolla on mahdollisuus yhdistää verkon solmussa eri suunnista tulevia paketteja yhdeksi uudeksi paketiksi. Koodauksen toimintaa on havainnollistettu taulukkoon 2 kuvatulla langattomalla verkolla. Solmut A ja C eivät voi suoraan lähettää tosilleen, mutta ne ovat solmun B kuuluvuusalueella. Solmu A lähettää viestin a solmulle C ja solmu C viestin c solmulle A. Kumpikin lähetys kulkee solmun B kautta. Viestien a ja c sijasta solmu B lähettää viestin $a \oplus c$ ($\oplus =$ XOR-operaatio), minkä A ja C kuulevat samaan aikaan. Koska kummallakin solmulla on vielä tallessa oma viestinsä, saa A selville C:ltä tulevan viestin yksinkertaisesti laskemalla $c = a \oplus (a \oplus c)$. Näin B saa perille kaksi viestiä yhdellä verkkolähetyksellä.

XOR-operaation sijaan voidaan käyttää sopivaa lineaarista koodausta, esimerkiksi äärellisen kunnan $\mathbb{GF}_{2^8}$ alkioita [FBW06]. Koodin valinnassa voidaan käyttää laskennaltaan kevyitä ja hajautettuja algoritmeja [WJMF05, FBW06, EKM07]

Koodauksesta saatava hyöty kasvaa verkon ruuhkaisuuden lisääntyessä eli mitä useamman solmun liikennettä on välitettävänä [KRH+06]. Koodaus lisää myös verkon toiminnan reiluutta, sillä ilman sitä paljon lähettävä solmu hyvän yhteyden takana voi kaapata koko verkon kapasiteetin toisten nälkiintyessä (engl. capture effect), mutta koodauksessa jokaisen solmun liikenne on samanarvoista.



Kuva 14: Probabilistiseen reititykseen verrattuna verkkokoodauksessa kaikki solmut saavat kaikki viestit (delivery ratio $= 1$) pienemmällä pakettimäärällä. Vaaka-akselilla (forwarding factor) on viestikopioiden tuottamisen todennäköisyys: arvolla 1 jokaisesta viestistä tehdään jokaisessa solmussa ylimääräinen epideeminen kopio [WB05].

Verkkokoodauksen on havaittu simulaatioissa lisäävän verkon suorituskykyä ja vikasietoisuutta etenkin harvoissa verkoissa, joissa on liikkuvia solmuja. Kuvan 14 kaaviossa on pystyakselilla kaikki viestit saaneiden solmujen osuus verkon kaikista solmuista (engl. packet delivery ratio) ja vaaka-akselilla ylimääräisten epideemisten viestikopioiden lähettämisen todennäköisyys (engl. forwarding factor). Simuloidussa sadan solmun verkossa verkkokoodaus saa toimitettua viestit kaikkiin solmuihin, vaikka viesteistä tehdään ylimääräisiä koodilohkoja vain todennäköisyydellä 0.125. Probabilistisella reitityksellä näitä epideemisiä viestikopioita pitää tuottaa vähintään todennäköisyydellä 0.75, ennen kuin kaikki viestit leviävät suurella todennäköisyydellä verkon kaikille solmuille [WB05].

3.6 Reitityksen yhteenveto

DTN-verkkojen reitityksessä otetaan laskelmoitu riski siitä, miten suurella todennäköisyydellä viestin halutaan menevän perille. Toimiva reititysstrategia on kompromissi kustannuksien ja toimitusviiveen välillä. Tässäkin suhteessa viivesietoiset verkot eroavat perinteisissä verkoista, joissa viestien oletetaan aina menevän perille.

Suoraviivaisin tapa saada aikaan parempia reitityspäätöksiä ja nopeuttaa viestien kulkua verkon läpi on lisätä reitityksessä käytettävän tiedon määrää tai laatua. Käytännössä tämä tarkoittaa tarkempaa tietoa tulevista kontakteista, joko varman tiedon pohjalta tai ennustamalla niitä paremmin [JFP04, BGJL06]. Tämän ratkaisun käyttökelpoisuus riippuu suuresti sovellusalueesta. Planeettojen liikkeitä tai satunnaisia radiohäiriöitä ei voida ennustaa nykyistä paremmin, mutta esimerkiksi liikkuvien käyttäjien ryhmäytymisen mal-

linnuksessa on vielä paljon tehtävää [ZZ07].

Verkkokoodausta on ehdotettu ratkaisuksi viivesietoisten verkkojen reititykseen ja viestivälitykseen [WB05, FBW06, KRH+06, EKM07, LLL07]. Verkkokoodauksessa onkin paljon haasteellisiin verkkoihin sopivia ominaisuuksia, kuten vikasietoisuus, pieni yleisrasitus sekä verkkokaistan optimaalinen käyttö.

Viivesietoisten verkkojen tyypillisten käyttötapauksien ajatellaan kuitenkin olevan tilanteita, joissa verkkokoodaus ei pääse tehostamaan liikennettä. Harvassa ja yhteyksiltään epäluotettavassa langattomassa verkossa ei solmulla ole välttämättä edes yhtä naapurisolmua radiokantaman päässä. Vaikka koodauksella ei tällöin suoraan paranneta verkon suorituskykyä, sitä halutaan ehkä kuitenkin käyttää vikasietoisuuden lisäämiseksi [LLL07, KRH+06]. Tähän soveltuvat ehkä paremmin pelkästään lähettäjän ja vastaanottajan välillä tehtävät poistokoodaus ja suihkulähdekoodaus, kuin verkon jokaista solmua koskeva verkkokoodaus.

4 DTN-verkkojen tulevaisuus

Edellisissä luvuissa esitetyllä DTN-arkkitehtuurilla voidaan toteuttaa sovelluksia ympäristöihin, joissa perinteiset verkkosovellukset eivät voi toimia [CBD+07]. Sen tavoitteena voidaan nähdä tietoliikenteen palvelumalli, joka pystyy tarjoamaan viestin välittämisen lähettäjältä vastaanottajalle hyvin erilaisilla toimitustavoilla ja siirtotekniikoilla.

Näin voidaan saada aikaan OSI-mallia täydentävä verkkosovellusten arkkitehtuuri, jonka käyttämisestä hyvillä verkkoyhteyksillä ei ole mitään haittaa, mutta joka huonoissa olosuhteissa toimii ratkaisevasti nykyisiä tekniikoita paremmin. DTN-verkkojen tutkimuksessa voidaan nähdä aineksia vallankumoukseen verkkojen ja niitä käyttävien sovellusten suunnittelussa ja toteutuksessa. Tässä luvussa tarkastellaan viivesietoisten verkkojen uusimpia tutkimustuloksia sekä niiden sovellusalueiden tulevaisuudennäkymiä.

4.1 Aktiivisia tutkimusalueita

Aktiivisimpia DTN-verkkojen tutkimusalueita ovat edelleen tehokkaiden reititystekniikoiden etsiminen, ryhmälähetyksen haasteet sekä siirtokerroksen toiminta [ZZ07]. Teoreettisen tutkimuksen lisäksi kehitetään myös työkaluja, kuten simulaattoreita, liikkuvuusaineistojen monipuolisempia hyödyntämistapoja sekä opportunistisen reitityksen testiverkkoja.

Reititys

DTN-verkkojen reitityksen ongelmiin löydetään erilaisia ratkaisuja sen mukaan, mitä sovellusaluetta kulloinkin mallinnetaan. Suuri osa ehdotetuista protokollista pyrkii ennustamaan solmujen välisiä yhteyksiä tapahtuneen historian perusteella [LDS03, SPR04a, BBL05, BGJL06]. Todellisiin liikkuvuusaineistoihin perustuvissa tutkimuksissa ei ole huomattu historiapohjaisten protokollien lämmittelyajasta olevan juurikaan hyötyä [SK07]. Teoreettisilla liikemalleilla lämmittely tuottaa hyötyä, mutta todellisissa verkoissa solmujen liikkeissä esiintyy alusta alkaenkin riittävää säännöllisyyttä. Todellisissa verkoissa on aina anomalistisia käyttäjiä, joiden liikkeitä ei voida ennustaa. Verkon pitäisi toimia tyydyttävästi myös heidän kannaltaan [SKJH03]. Tällöin käytetään yleensä epideemistä reititystä tai koodaukseen perustuvia ratkaisuja [WJMF05, WB05, LFC05].

Epideeminen reititys ei tarvitse mitään ennakkotietoa verkosta, joten se on yleiskäyttöisin, yksinkertaisin toteuttaa ja toimitusviiveiltään paras. Ehdotettu PREP-protokolla (PRioritized EPidemic) pienentää epideemisen reitityksen resurssikulutusta välittävien ja pudotettavien viestien jonoja hallitsemalla [RHB+07]. PREP pyrkii pitämään solmujen viestipuskurit mahdollisimman täynnä, jotta viesteillä on mahdollisimman suuri todennäköisyys päästä perille.

PREP-protokollassa jokainen solmu ylläpitää koko verkon topologiatietoa link state -algoritmilla MEED-protokollan tapaan [JLW05] ja lisäksi arvioi naapureidensa välityskykyä yhteyshistorian avulla. Viesteissä on tieto siitä, miten monen solmun kautta ne ovat kulkeneet, sekä lähetys- ja voimassaoloajat. Näiden tietojen perusteella solmut voivat valita, missä järjestyksessä ne vaihtavat viestejä sekä mitkä viestit pudotetaan verkosta. Ensimmäisenä lähetetään viestit, jotka ovat lähimpänä määränpäätään eli joiden hukkumisessa menetettäisiin eniten jo tehtyä työtä. Vastaavasti viestejä aletaan poistaa verkosta, vasta kun ne ovat tarpeeksi kaukana lähettäjästä, jolloin niillä on parempi mahdollisuus päästä perille. PREP-protokolla ei yritä minimoida verkon tai talletustilan käyttöä. Se ei helposti hukkaa viestejä ruuhkautumisen vuoksi, koska edellä kuvattu viestien lähetys- ja poistojärjestys suosii viestien tehokasta etenemistä.

PREP toimii selvästi tehokkaammin kuin epideeminen reititys etenkin harvoissa verkoissa, joissa yhteydet ovat satunnaisia ja kun solmujen talletustila on rajoitettu. Verkon topologiatiedon ylläpitäminen tuottaa hallinnollista liikennettä, mutta sen määrää voidaan hallita halutun riskitason mukaan asettamalla sopiva hystereesi muutoksien levittämiselle MEED-protokollan tapaan. Topologiatiedon levityksen luotettavuutta voitaisiin parantaa myös koodaustekniikoilla.

Useissa DTN-verkkojen sovelluksissa viestejä välittävät dedikoidut välityssolmut, vies-

tilautat. Viestilauttojen lukumäärän tai reittien valinta tietyillä kriteereillä ovat haastavia tutkimusongelmia. Yhden viestilautan käytössä ei hukata resursseja, mutta se ei ole kovin vikasietoista. Riittävän laajassa verkossa tarvitaan useampia lauttoja [ZZ07]. Verkossa voidaan käyttää myös erillistä pitkän kantaman radiota hallintaliikenteeseen esimerkiksi lautan aikataulujen tai solmun lähetystarpeiden ilmoittamiseen.

Aikataulupohjaisten reittimallien vertailu reittejään ajavien bussien keskinäisten kohtaamisen aineistosta on paljastanut, että solmujen välisten kohtaamisien lisäksi on syytä mallittaa tiettyjen reittien eli liikkuvuusmallien välisiä yhteyksiä [ZKL+07]. On siis parempi mallittaa samaa reittiä kulkevien solmujen kohtaamista, kuin kaikkien solmujen kohtaamista.

Ad hoc -verkkojen ja DTN-verkkojen reititysprotokollien yhdistämistä on ehdotettu toteutettaviksi siten, että sovellukset voivat valita, toimivatko ne erilaisissa verkko-olosuhteissa päästä päähän -semantiikan vai DTN-semantiikan mukaan. Ehdotetuilla AODV-protokollan laajennoksilla reittihaun yhteydessä saadaan selville tarjolla olevat DTN-reitittimet ja -palvelut [OKD06]. Vaikka AODV ei löytäisikään yhtenäistä reittiä kohteeseen, voi viivesietoinen sovellus toimia koko ajan parhaalla verkkoyhteyden sallimalla tavalla (engl. always best connected).

Ryhmälähetys

Pelastussovelluksissa käytettävät DTN-verkot tarvitsevat luotettavaa ryhmälähetystä (engl. multicast) esimerkiksi vaara-alueista tiedottamiseen tai potilastietojen välittämiseen. Perinteisten verkkojen ryhmälähetyksessä ei ryhmän kokoonpanon muutoksiin viestinvälityksen aikana tarvitse varautua, mutta DTN-verkoissa niitä tapahtuu paljon todennäköisemmin. Ryhmälähetyksen ja ryhmäjäsenyyden semantiikat tarvitsevat DTN-verkkoihin uudet määritelmät [AS06].

Ehdotetut ratkaisut jättävät jäsenyyden määritelmän sovelluksen päätettäväksi [ZAZ05]. Viestissä voidaan määritellä, että viestin vastaanottajia ovat ne solmut, jotka ovat kuuluneet ryhmään tietyllä aikavälillä (Temporal Membership, TM), tai että lisäksi viesti toimitetaan tiettynä aikavälillä (Temporal Delivery, TD), tai että vielä edellisten lisäksi solmut ovat ryhmän jäseninä toimitushetkellä (Current-Member Delivery, CMD). Näistä kahdessa ensimmäisessä riittää, että solmu on ollut ryhmän jäsen joskus kyseisinä ajankohtina. Viimeisin määritelmä vastaa eniten perinteistä monilähetyksen semantiikkaa, jossa solmu on ryhmän jäsen viestin lähetys- ja toimitushetkellä.

Vielä ryhmälähetystäkin haasteellisempaa on jokulähetyksen (engl. anycast) toteuttaminen DTN-verkkoihin. Jokulähetyksessä viesti toimitetaan tietyn solmuryhmän jollekin,

mieluiten vain yhdelle, vastaanottajalle. Myös jokulähetyksen toteutus DTN-verkoissa tarvitsee uutta ryhmäjäsenyyden mallia [GXZ+06].

Kuljetuskerroksen toiminta

Kuljetuskerroksen luotettavan toiminnan, vuonvalvonnan ja ruuhkanhallinnan ongelmia tutkitaan edelleen muita DTN-verkkojen haasteita vähemmän [HABR05, Zha06, BJS06, ZZ07]. Toimitusvalvonnan siirto ratkaisee osan luotettavuusvaatimuksista, mutta siinäkin on ongelmia osittuneiden verkkojen yhtyessä ja käytettäessä useampia viestikopioita ja toimitusvalvojia. Useamman valvojan tapauksessa voidaan käyttää poistokoodausta tai ennakoivaa fragmentointia vikasietoisuuden lisäämiseksi [SFM06].

Kahden solmun välisellä yhteydellä voidaan käyttää tehokkaan virheenkorjauksen sisältävää sarjallista Licklider Transmission Protocol -protokollaa (LTP), joka tosin suuriin siirtoviiveisiin kehitettynä vaatii käytettävien parametrien sopimisen ennakolta [RBF07]. LTP-protokollasta on ideoitu yleisempään käyttöön oma versionsa, mutta sen kehittely on vielä kesken.

DTN-reitittimiin on ehdotettu itsellistä ruuhkanhallintaa (engl. autonomous congestion control), jolloin ne voivat suojautua liialliselta liikenteeltä kuitenkaan kadottamatta viestejä [ZZ07]. Solmujen talletustilaa voidaan vapauttaa käyttämällä erilaisia kuittausmenettelyjä, vaikka niistä syntyykin lisää liikennettä [HABR05]. Solmujen resurssikulutuksen hallintaan on ehdotettu ekonomiseen hinta- ja riskianalyysiin perustuvaa reititysmallia [BJS06].

Tutkimusmenetelmien kehitys

Teoreettisen tutkimuksen ja simulaatioiden lisäksi DTN-verkkoihin ehdotettuja ratkaisuja on testattu todellisista verkoista kerätyillä liikkuvuusaineistoilla ja pienillä testiverkoilla. Näin tehdyt simulaatiot käyttävät todellisista verkoista kerättyjä kontaktitietoja mallintaessaan protokollien toimintaa. Näin niiden tulokset ovat luotettavampia, kuin jos kontaktit generoidaan jostakin teoreettisesta liikemallista.

Yleisesti käytetty *ns-2*-simulaattori on suorituskyvyltään riittämätön suurien verkkojen simulointiin [SK07]. Muutkaan simulaattorit eivät yleensä mallita luotettavasti esimerkiksi suuria siirtoviiveitä [WPP+07]. DTN-ympäristöjen simulointiin erikoistuneilla simulaattoreilla, kuten *dtnsim* ja *ONE* voidaan paremmin tutkia DTN-verkkojen ominaispiirteitä [Ker07]. *ONE* sisältää myös visualisointinäkymän solmujen liikkeiden ja niiden kuljettamien viestien havainnollistamiseksi.

Langattomien laitteiden liikkeistä on saatavilla laajoja aineistoja useiden vuosien ajalta [SK07]. Niillä voidaan tutkia ehdotettujen DTN-reititysprotokollien toimintaa suurilla solmumäärillä ja realistisilla kontaktiprofiileilla. Saadut tulokset ovat olleet yhteensopivia simulaatiotulosten kanssa. Tällä tavalla voidaan myös tutkia, millä tavalla solmujen liikettä pitää mallittaa, jotta simulaation tuloksena on todellisuudessa havaittu reitityksen suorituskyky [ZKL+07].

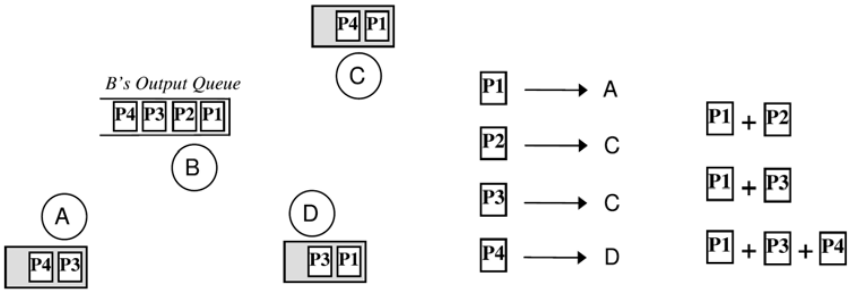
Opportunistiset testiverkot

Joissakin verkkosovelluksissa voitaisiin jo nyt hyödyntää erillistä opportunististen kontaktien havaitsemiseen tarkoitettua dedikoitua elektroniikkaa. Näin verkkosolmu voi torkkua ja säästää tehokkaasti virtaa, mutta herätä hyödyntämään ohi kulkeva viestilauttaa. Tällaisia laitteita on jo suunniteltu sensoriverkkoihin, mutta niitä voitaisiin hyödyntää muissakin langattomissa verkoissa [HYW+06].

Opportunistisessa reitityksessä (engl. opportunistic routing/forwarding) hyödynnetään langattoman siirtotien yleislähetystä (engl. broadcast) sekä erilaisia koodaustekniikoita. Sen avulla vähennetään myös siirtohäiriöiden vaikutusta. Käytössä olevissa hajaverkkotekniikalla toteutetuissa kaupunkiverkoissa on yhteyksiä, joissa siirtohäiriöiden osuus on jopa 30 % siirtokaistasta [CJKK07].

Opportunistisesta verkkokoodauksesta on tehty Linux-alustalle avoimen koodin prototyyppitoteutukset COPE (coding opportunistically) [KRH+06] ja MORE (MAC-independent opportunistic routing protocol) [CJKK07]. Niiden toiminta perustuu sovelluksille tuntumattomaan (engl. transparent) koodausmahdollisuuksien tunnistamiseen langattomassa verkossa. COPE yhdistää eri solmujen välisiä tietovoita, MORE myös samasta solmusta lähtevän tietovuon eri paketteja. Seuraavassa käsitellään esimerkkinä COPE-toteutuksen toiminta.

COPE-tekniikassa verkon solmut ottavat vastaan kaiken langattoman verkon liikenteen ja keräävät puskuriin verkossa kuulemansa paketit. Samoin kaikki lähettäminen verkkoon tehdään yleislähetyksenä. Solmut kuuluttavat ajoittain varastoimiensa pakettien tunnisteet ja oppivat verkkoa kuuntelemalla toisten solmujen puskurimuistien sisällöt. Kun solmulla on omia tai välittämiään paketteja lähetettävänä, se tarkistaa, mitä paketteja naapurisolmuilla on muistissaan. Sen perusteella solmu valitsee sellaisen pakettien XOR-yhdistelmän, jolla mahdollisimman moni lähetys saadaan perille. Tässä siis hyödynnetään langattoman verkon ominaisuutta lähettää kerralla viestejä usealle solmulle.



(a) Solmujen A, C ja D pakettipuskurei- den sekä solmun B lähetysjonon sisällöt.

(b) Solmun B lähetysjonon pakettien kohteet.

(c) Solmun B eri koodausmahdollisuudet. Viimeinen vaihtoehto on tehokkain.

Kuva 15: COPE onnistuu lähettämään yhdellä kertaa kolmen viestin vastaanottoon tarvittavan tiedon [KRH+06].

Kuvassa 15(a) esitetään solmun B lähetysjonon sisältö sekä muiden solmujen pakettipuskurien sisällöt. Koska solmu B tietää, mitä paketteja verkon toisilla solmuilla on, ja mihin solmuihin sen varastoimat paketit ovat kuvan 15(b) mukaan menossa, se voi yhdistää paketit P1, P3 ja P4 kuvan 15(c) alimman vaihtoehdon mukaan ja saada perille kolme pakettia yhdellä verkkolähetyksellä.

Verkkokoodaus tuottaa sitä enemmin hyötyä, mitä enemmin paketteja on tarjolla koodattavaksi. COPE tehostaa liikennöintiä eniten, kun verkossa on useita riippumattomia UDP-yhteyksiä solmujen välillä. Tällöin verkon suorituskyky jopa kolmin-nelinkertaistuu verrattuna koodaamattomaan ratkaisuun. Langattomassa monihyppyverkossa piilolähettäjien häiriöiden vuoksi TCP ei lähetä tarpeeksi paketteja, joiden läpimenoa voitaisiin tehostaa koodaamaalla [KRH+06]. Ilman piilolähettäjiä TCP-yhteyksien nopeutus olisi 38 %. Yhteydet ulos hajaverkkotekniikalla (engl. mesh network) toteutetusta langattomasta verkosta yhden reitityspisteen kautta nopeutuvat 5-70 % sisään tulevan ja ulos menevän liikenteen suhteen mukaan. Koodauksen tehostus kasvaa verkon ruuhkautuessa, joten se vähentää ruuhkautumisen vaikutuksia.

Verkko- ja kuljetuskerroksien heikko yhteistoiminta langattomissa IEEE 802.11-verkoissa saa aikaan epäsymmetriaa datan ja sen kuittauksien välillä [PWWM05, FLZ+05]. COPE helpottaa tätä vuointerferenssiä, koska sen avulla välittävä solmu, esimerkiksi tukiasema, saa välitettyä enemmin liikennettä jokaisella lähetysvuorollaan.

COPE toimii verkko- ja siirtokerroksien välillä, joten sen toiminta on koneiden sovelluksille täysin läpinäkyvää COPE ei hidasta verkon toimintaa, sillä se ei pysähdy odottamaan

lisää koodattavaa, jos sitä ei paketin lähetyshetkellä ole saatavilla.

Myös MORE nopeuttaa liikennettä vastaavilla tavoilla. Lisäksi se toimii paremmin kuin COPE tilanteissa, joissa tietovoita on vain vähän, sillä se koodaa satunnaisesti myös saman vuon sisäisiä viestejä [CJKK07]. Verkkokoodauksen yleisrasite ei hyvin valituilla koodeilla ollut testatussa 20 koneen hajaverkossa liian suuri. Lähetetyn liikenteen määrä oli 5 % suurempi kuin koodaamattomassa tapauksessa.

Verkkokoodaus muuttaa niin perinpohjaisesti informaation välittämistä verkossa, että sen omaksuminen vienee aikaa. Toisaalta sen edut langattomissa verkoissa ovat kiistattomat ja esimerkkitoteutuksien COPE ja MORE kokemuksien mukaan myös helposti saavutettavat. Puhelinvalmistaja Ericsson on toteuttanut opportunistista verkkokoodausta viestivirtojen lähettämiseen [LJ06]. Koodaustekniikat ovat myös tehokas tapa toteuttaa erilaisia palveluita, joiden toteuttamiseen on tähän tarvittu multicast-lähetteitä [GR05].

Analoginen verkkokoodaus

Uusimmat tutkimukset osoittavat, että verkkokoodaus voidaan toteuttaa myös analogisena eli fyysisellä tasolla (engl. analog network coding) [KGK07]. Perinteiset langattomien verkkojen MAC-protokollat ovat pyrkineet välttämään lähetyksien välisiä törmäyksiä. Analogisessa verkkokoodauksessa lähetteiden annetaan sekoittua ja välittävä reititin voimistaa syntynyttä häiriösignaalia. Koska kummallakin lähettäjällä on tiedossa oma alkuperäinen lähetteensä, ne voivat poistaa vastaanottamastaan häiriöisestä signaalista oman lähetteensä ja saada selville vastaanotettavan lähetteen. Toiminta on täsmälleen samanlaista kuin perinteisessä verkkokoodauksessa, mutta lähetteiden sulauttaminen tapahtuu jo lähetysvaiheessa.

Ohjelmistopohjaisilla radioilla tehty implementaatio osoitti, että analoginen verkkokoodaus toimii, vaikka siirtotiessä on satunnaisia häiriöitä ja vaikka lähetteitä ei ole mitenkään synkronoitu keskenään. Sopivissa topologioissa koodaus poistaa täysin piilolähettäjäongelman ja lisää siirtotien rinnakkaiskäyttöä (engl. spatial reuse).

Analogisen verkkokoodauksen toimivuus on osoitus tarpeesta kehittää uutta langatonta tietoliikennetekniikkaa, joka ei perustu tuttujen menetelmien triviaaleille nopeutuksille ja laajennoksille. Osa langattoman siirtotien ongelmista voidaan innovatiivisella tekniikalla kääntää sen eduksi. Näiden tekniikoiden kehittämisessä tarvitaan uudenlaista tietojenkäsittelytieteen ja tietoliikennetekniikan tutkimusyhteistyötä.

4.2 Uuden Internetin sovellukset

Langattomien tekniikoiden ja liikkuvien käyttäjien yleistymisen vuoksi ollaan määrittelemässä verkkosovellusten arkkitehtuuria, joka esimerkiksi kestää paremmin viiveitä ja häiriöitä [Fall06, Ott06, SGP+07]. Muita suunnittelukriteereitä ovat esimerkiksi turvallisuuden ja itsekorjaavuuden toteuttaminen [SGP+07].

Moni sovellusprotokolla vaatii päästä päähän -yhteyttä palvelimelle asti, vaikka itse sovelluksen toiminta etenkään käyttäjän kannalta ei sitä edellyttäisikään [Ott06]. Sähköpostin asiakasohjelma epäonnistuu sähköpostin lähettämisessä, jos se ei saa lähetyshetkellä yhteyttä SMTP-palvelimelle. Käyttäjän kannalta riittäisi, että ohjelma ottaa viestin vastaan ja toimittaa sen perille, kunhan verkkoyhteys on saatavilla. Samalla tavalla WWW-sivun nouto voisi käyttäjän kannalta hyvin kestää muutamia sekunteja pidempään siksi, että sivu välitetään verkossa vaiheittain useamman solmuvälin yli. Käytännössä tätä ei voida toteuttaa, koska HTTP-protokolla ei toimi ilman TCP-yhteyttä [OK06].

Perinteiset liikkuvien käyttäjien verkkosovellukset rakennetaan arkkitehtuurilla, jossa käyttäjän liikkuvuus, verkon toteutuksen yksityiskohdat tai yhteyden saatavuus piilotetaan kokonaan sovellukselta. Tällä tavalla toimii esimerkiksi Mobile IP [Per02] tarjotessaan sovelluksille näkymän koko ajan käytettävissä olevasta yhteydestä.

Liikkuvan käyttäjän tapauksessa kiinteän Internet-yhteyden matkiminen sovelluksien vuoksi epäonnistuu kuitenkin ennen pitkää. Tällöin vaatimattomammallakin yhteydellä toimivien sovellusten käyttö estyy tavallaan turhaan. DTN-arkkitehtuurin avulla näistä sovelluksista saadaan vikasietoisempia. Palvelinkeskeisen toimintatavan (engl. server centric, client-server) rinnalle määritellään myös solmuväli kerrallaan toimiva liikennöintitapa (engl. hop-by-hop). Sovelluksien pitää toki osata käsitellä yhteyskatkoja ja muita ongelmia, mutta näin esimerkiksi jotkut henkilökohtaisen viestinnän sovellukset voivat toimia käyttäjien kannalta odotetusti [SAP, MO06]. Jatkuvaa yhteyttä vaativia interaktiivisia sovelluksia ei kuitenkaan voida toteuttaa tällä tavalla.

DTN-tekniikoiden yleistyminen

DTN-tekniikat tulevat käyttöön todennäköisesti ensimmäisenä erikoissovelluksissa, joihin ei vielä ole käytössä valmista arkkitehtuuria tai laajassa käytössä olevia sovelluksia. Helpointa käyttöönotto on uusilla sovellusalueilla, kuten syrjäseutujen verkkoyhteyksissä, joissa ei tällä hetkellä ole käytössä valmiita ratkaisuja. Sensoriverkkojen ja avaruustutkimuksen piirissä DTN-protokollien pitää aluksi syrjäyttää nykyiset ohjelmistot.

Suurten käyttäjämäärien palveluiksi DTN-protokollilla on pidempi tie. Totuttu Mooren laki ja elektroniikan kehittyminen tuottanevat ennen pitkää riittävän langattomaan tiedon-

siirtoon pystyvän laitekannan. Olennaiset kysymykset liittyvätkin ad hoc -tyyppisen verkkokäytön yleistymiseen ja muutokseen verkkoyhteyksien toteutus- ja tuottamismalleissa. Nykyisestä voimakkaasti infrastruktuuripohjaisista, palveluntuottajien hallitsemista yhteyspalveluista siirrytään käyttäjien keskenään muodostamiin langattomien verkkojen yhteisöihin [MRPS04].

Näin esimerkiksi älypuhelimen verkkoyhteys Internetiin ei enää ole puhelinliittymän toteuttama palvelu, vaan se saadaan aikaan verkostoitumalla lähistön muiden langattomien laitteiden kanssa siinä toivossa, että jollakin niistä on käytössään riittävä Internet-yhteys. Verkkoyhteyden saatavuus perustuu riittävän suureen käyttäjämassaan samaan tapaan, kuin tietty tiedosto on aina saatavilla riittävän suuresta vertaisverkosta. Saatavan yhteyden nopeus ja laatu eivät välttämättä riitä ääni- tai kuvapuheluihin, mutta esimerkiksi pikaviestintäsovelluksien käyttö voi onnistua normaalisti. BITNET-verkon käyttäjän kannalta reaaliaikaiselta vaikuttava pikaviestinsovellus oli toteutettu jo 1980-luvulla siirtämällä tiedostoja solmuväli kerrallaan [Ste], joten viestinvaihdon semantiikalla toimiva DTN-sovellus olisi helppo toteuttaa.

Tällaisten palvelujen toteuttaminen ei ole palveluntarjoajien ja teleoperaattoreiden etujen mukaista. Avainasemassa ovatkin laitevalmistajat, joiden on saatava aikaan edullisia, mutta tarpeeksi monipuolisia ad hoc -verkkoja hyödyntäviä laitteita. Toinen avainkysymys ovat aktiivikäyttäjäfoorumit, jotka voivat tällaisia sovelluksia toteuttaa ja organisoida. Analogia tälle mallille voidaan nähdä vertaisverkkojen syntyhistoriassa. Tiedostonvaihtoon tarkoitetut vertaisverkot alkoivat toimia, kun riittävän monella kotikäyttäjällä oli käytössään kone nopean verkkoliittymän päässä.

Tällä tavalla muodostettujen palveluiden käyttäminen ei kuitenkaan ole loppukäyttäjän kannalta ongelmatonta. Uskalletaanko tällaisen verkon varaan jäädä ja luotetaanko välittävien solmujen yhteistyöhalukkuuteen? Osaan näistä riskeistä voidaan varautua esimerkiksi suojaustekniikoilla tai hankkimalla varmistusyhteydeksi kaupallinen verkkoliittymä. DTN-reititysmallit osaavat toimia erilaisilla ja erihintaisilla verkkoyhteyksillä, joten käyttäjän kannalta DTN-tekniikka auttaa hallitsemaan näitä riskejä [MO06].

DTN-verkkojen tietoturva ja tietosuoja vaativat vielä tutkimusta, mutta esimerkiksi tiedostojen tai sähköpostikansioiden etäkäytön suojaus voidaan toteuttaa nykyisellä tekniikalla ongelmitta ja käyttää DTN-tekniikkaa vain tapahtumatietojen välittämiseen. Yksittäisen solmun verkolle antamien välityspalveluiden määrälle voidaan antaa sopivaksi katsottu yläraja, jolloin solmu on suojassa verkkoliikenteen rasituksilta. Koko verkon suojakysymyksiä ja suorituskyvyn turvaamisen ongelmia ei näin vielä ratkaista.

DTN-sovellusrajapinnan edut

Verkkosovelluksen ja verkon välisen rajapinnan rikastuminen toisaalta helpottaa, toisaalta vaikeuttaa sovellusten kehittämistä. Suurin haaste on koko sovellusarkkitehtuurin muuttuminen läpinäkyväksi alla olevan verkon suuntaan. DTN-sovellukset eivät enää voi luottaa alla olevan yhteyden tarjoamaan virheettömään päästä päähän -yhteyteen [OK06].

DTN-sovelluksien käyttössä on suora rajapinta verkkoliikenteen palvelunlaatuparametrien ja eliniän hallintaan, mitä ei vanhassa OSI-sukuisessa arkkitehtuurissa ole ollut mahdollista. Palvelunlaadulle määritellyt kolme tasoa ovat ehkä hiukan vähän, mutta uusien määrittely standardiin ei ole vaikeaa. Yleistyessään DTN-arkkitehtuurilla on hyvät mahdollisuudet tarjota verkkosovellusten palvelunlaadun hallinnan vakioratkaisu, jota voidaan soveltaa hyvin monenlaisiin käyttötapauksiin. DTN-sovelluksiin mukautetut LTP- ja Saratoga-protokollat sisältävät jo näitä palvelunlaadun määritteitä sovellusrajapinnassaan [RBF07, WEI+07].

DTN-tekniikalla vikasietoisuutta

DTN-tekniikan vahvuudet ovat sen vika- ja viivesietoisuudessa. Kärjistetysti voidaan sanoa, että jos epideeminen DTN-reititys ei saa viestiä perille, ei siihen pysty mikään muukaan protokolla. Vielä lisää vikasietoisuutta saadaan yhdistämällä DTN-viestinvälitys erilaisiin koodaustekniikoihin.

DTN-tekniikalla voidaan toteuttaa erilaisia hätä- ja turvayhteyksiä myös kuluttajalaitteisiin. Periaatteessa olisi mahdollista, että kaikkiin langattomilla yhteyksillä varustetuihin laitteisiin toteutetaan hätäyhteysmoodi, jossa ne verkottuvat keskenään ja välittävät esimerkiksi hätä- ja viranomaisviestejä tai toimivat toisilleen tukiasemina. Näiden yhteyksien suojaaminen häirinnältä ja väärinkäytöksiltä vaativat toki käytännön ongelmien ratkaisua, mutta esimerkiksi suuronnettomuuksissa tällaisista ominaisuuksista olisi kiistatta hyötyä. Tällaisten ratkaisujen toteutuminen vaatii myös viranomaistahojen osallistumista kehitykseen.

Samojen vikasietoisuusominaisuuksien vuoksi DTN-tekniikka voi yleistyä myös osana infrastruktuuriverkkojen toteutusta. Esimerkiksi verkon aktiivilaitteiden verkonvalvonta ja -hallintaliikenteen turvaaminen kaikissa tilanteissa voisi hyötyä erittäin vikasietoisesta verkkoliikenteestä, joka osaa käyttää hyväkseen eri tekniikoilla toteutettuja yhteyksiä. Tästä jonkinlainen prototyyppiratkaisu on seismografiverkon hallintaliikenteen toteutus DTN-tekniikalla [LGE06].

Verkonvalvonnan perusprotokolla SNMP[CFSD90] on UDP-pohjainen verkkoyhteydetön protokolla, joten sen päällä toimivat sovellukset voisivat hyvin käyttää DTN-arkkitehtuu-

rin palveluita ilman suuria muutoksia. Samaa sovitusta voitaisiin käyttää myös SNMP-protokollan korvaajaksi suunnitellun NETCONF-protokollan kanssa [Enn06, CGG+04].

Nykyiset DTN-verkot

Aktiivisesta tutkimuksesta huolimatta DTN-tekniikoilla ei ole vielä toteutettu todellisia palveluverkkoja. Referenssi-implementaation [DTNRG] lisäksi DTN-arkkitehtuurin osia on toteutettu Symbian-älypuhelimille [DASM, HKL+07] sekä joihinkin avaruussovelluksiin [WEI+07]. Pieniä DTN-tyylisiä sensoriverkkoja on toteutettu ympäristöseurantaan ja turvallisuusvalvontaan [LWW06, DVBJ04]. Nämä kaikki ovat vielä enemmin teknologia- ja protokollademonstraatioita, kuin valmiita sovelluksia. Toisaalta, paljon pidempään tutkittuja ad hoc -verkkojakaan ei vielä ole tuotannossa kovinkaan monia [KM07].

DTN-verkkojen tapaan toimiva United Villages -verkko yhdistää jo satoja Aasian maaseutukyliä Internetiin bussien, jakeluautojen ja moottoripyörien kuljettamilla viestisolmuilla. Palvelun toimittava First Mile Solutions on laajentamassa verkkoa 220 000 kylään koko Intiassa [Bas07].

United Villages tarjoaa esimerkiksi ääniviestit, tekstiviestit, sähköpostin ja WWW-haut vuorokauden toimitusajalla sekä elektronisen kaupankäynnin peruspalvelut [DAKNET]. Vaikka toteutus ei käytä virallista DTN-ohjelmistoa, vaan omia DakNet-tuotteitaan, ovat sen tekniset ratkaisut tärkeä käytännön referenssi myös DTN-tutkimukselle. Olosuhteet kuumissa ja pölyisissä maalaiskylissä ilman jatkuvaa sähkönjakelua täyttävät kyllä haasteellisten verkkojen tunnusmerkit [BDH+06].

4.3 Tutkijan haastattelu

DTN-verkkojen tutkimuksen ja sovellusten tulevaisuudennäkymistä haastateltiin alan aktiivista tutkijaa, professori Jörg Ottia Teknillisen korkeakoulun tietoverkkolaboratoriosta [NETLAB]. Seuraavassa on englanniksi sähköpostilla tehdyn haastattelun tiivistelmä. Koko haastattelu on liitteessä 1.

DTN-arkkitehtuuri

Reitityksen viisastenkiveä ei ole vielä löydetty. Epideeminen reititys on yksinkertainen toteuttaa, mutta se sopii vain joihinkin sovellusalueisiin. Reitityksen ongelmiin on löydettävä toimivia ratkaisuja ennen kuin DTN-sovelluksia voidaan käyttää suurissa verkoissa. Poistokoodauksen edut ovat resurssien hallinnassa ja toimitusaikojen parantamisessa. Verkkokoodauksen etuja ei taas päästä harvassa verkossa hyödyntämään sovellustasolla ja sen käytöstä siirtokerroksessakin tarvitaan lisää tutkimusta.

Verkon turvallisuuden ja väärinkäytön määritelmät ovat DTN-verkoissa erilaisia kuin perinteisissä verkoissa. Sovellustason turvallisuus voidaan toteuttaa päästä päähän -menetelmillä. Välittävien solmujen on sen sijaan vaikea päätellä, mikä viesti on laillinen ja mikä pitäisi pudottaa verkosta. Eräs tapa vähentää väärinkäytöksien vaikutusta on käyttää palveluluokkatunnisteita, joista ei voi päätellä niiden tuomaa hyötyä. Näin väärinkäytöstä yrittävä solmu ei osaa teeskennellä parempaa palvelua saavaa solmua.

Palvelunlaatumäärityksien mukanaolo DTN-arkkitehtuurissa ei vielä tarkoita, että mikään sovellus alkaisi käyttää niitä. Internetin omiakaan palveluluokkia ei vuosiin käytetty mihinkään, eivätkä ne vieläkään ole laajassa käytössä. Verkkopalvelun laatuluokkien käyttö tuntuu liioittelulta esimerkiksi verkossa, jossa vain 75 % viesteistä edes menee perille.

Sovelluksen ja verkon välisen rajapinnan rikastuminen antaa mahdollisuuden toteuttaa paremmin häiriöitä kestäviä sovelluksia niihin käyttötapauksiin, joissa häiriösietoisuutta kaivataan. Sovellusohjelman tekijä voi esimerkiksi valita synkronisen ja asynkronisen toimintatavan välillä.

Sovellukset

DTN-verkkojen tutkimus ei ole yrittänyt tarjota tuotoksiaan ratkaisuksi kaikkiin perinteisen tietoliikenteen ongelmiin. Tavoitteet ovat toki kunnianhimoisia, mutta ne perustuvat ainakin koettuun todelliseen tarpeeseen esimerkiksi sensoriverkkojen käyttäjien taholta. Tässä ollaan ehkä viisastuttu MANET-verkkojen kokemuksista, eikä yritetä kehittää satoja reititysprotokollia ilman todellisia sovellustarpeita.

Vaikka Internetiinkin on kehitetty uusia perusprotokollia, ei kaikkien verkkosovelluksien tarvitse perustua IP-protokollaan. Standardiehdotuksen DTN-sovellusmalli voi olla pienten laitteiden kannalta turhan raskas. Se saattaa silti olla riittävän hyvä yleisratkaisu haastavien verkko-olosuhteiden sovellusalustaksi. Toisaalta osa sovelluksista tulee kuitenkin vaatimaan omat kevyemmät ratkaisunsa, eikä esimerkiksi viivekriittisten sovellusten toimintaa ole järkevää rakentaa raskaalle viivesietoiselle verkkopalvelumallille.

Tällä hetkellä kehitettävän DTN-tekniikan elinkaarta tai sovellusalueiden määrää on vaikea ennustaa. Jotkut nykyisin ajatellut tarpeet DTN-tekniikalle saattavat poistua muun tekniikan kehittyessä, mutta esimerkiksi planeettainvälisen tietoliikenteen siirtoviiveet säilyvät jatkossakin suurina.

DTN-verkkojen vikasietoisuutta voidaan käyttää turvaamaan kriittistä liikennettä, esimerkiksi palvelumäärityksien tai ohjelmistojen päivityksissä huonojen yhteyksien takana oleviin laitteisiin. Tällöin DTN-viestintää käytetään osana infrastruktuuriverkkojen toteutusta. Aidompien ad hoc -ympäristöjen toteuttaminen esimerkiksi pelastussovelluksissa ta-

pahtuu todennäköisemmin tuomalla paikalle uutta kalustoa kuin käyttämällä olemassaolevia laitteita. Näistäkin sovellusalueista tarvittaisiin lisää käytännön kokemuksia.

Lopulliset sanansa DTN-tekniikan käyttökelpoisuudelle sanovat kuitenkin käyttäjät. Tekniikan pitää tuoda lisäarvoa ja olla luotettavaa, muuten sillä ei ole mahdollisuuksia yleistyä. Vaikka viivesietoiset ratkaisut löytänevätkin soveltajia erikoisverkkojen piiristä, on paljon vaikeampi ennustaa, mitä potentiaalisesti valtavat mobiililaitteiden käyttäjien yhteisöt voivat saada aikaan.

5 Yhteenveto

Maailmanlaajuisen Internetin perusprotokollat toimivat vain verkoissa, joiden siirtonopeudet, virhetiheydet ja kiertoviiveet vastaavat niitä etäisyyksiä ja tietoliikennetekniikoita, joiden ehdoilla protokollia on kehitetty. Suuret poikkeamat näissä parametreissa voivat estää protokollien toiminnan kokonaan tai ainakin merkittävästi heikentää niiden tehokkuutta.

Viive- ja häiriösietoiset verkot (Delay/Disruption Tolerant Networks, DTN) ovat tietoliikenneverkkoja, joissa siirtoviiveet ja virhetiheydet ovat Internetiä suurempia sekä yhteyskatkot tavallisia. Näissä verkoissa toimivat sovellukset ovat mukautuneet yhteyskatkoihin sekä päästä päähän -yhteyden puuttumiseen, joten niille riittää, että viesti toimitetaan perille esimerkiksi tiettyyn aikaan mennessä. Tässä tutkielmassa esitellään viivesietoisten verkkojen arkkitehtuurin perusratkaisuja sekä joitakin sovellusalueita. Erityisesti käsitellään DTN-verkkojen reitititystä sekä niissä käytettävien sovellusten haasteita.

DTN-verkkojen voidaan ajatella pyrkivän kattamaan olemassa olevien verkkotekniikoiden jättämät aukot. Toisin kuin Internetissä, jossa välitetään paketteja ja bittivirtaa jatkuvan verkkoyhteyden päällä, välitetään DTN-verkoissa sovellusten kannalta mielekkäitä, mahdollisesti hyvinkin suuria tietokokonaisuuksia, viestinippuja. Viestit siirretään verkossa mahdollisesti vain yksi solmuväli kerrallaan ja talletetaan välittäviin solmuihin.

Viivesietoisten verkkojen sovelluksia ovat esimerkiksi yhteydet avaruusluotaimiin toisilla planeetoilla tai viestinvälitys seuduilla, joilla ei ole tarjolla kiinteää tietoliikenneinfrastruktuuria. Muita sovellusalueita ovat pelastus- ja sotilasyhteydet, sensoriverkot sekä liikkuvien käyttäjien ja ajoneuvojen verkot.

Verkon jokainen solmu on myös DTN-reititin, perinteisten ad hoc -verkkojen tapaan. DTN-arkkitehtuurin viive- ja vikasietoisuus syntyy solmujen välisten yhteyksien opportunistisesta käytöstä, toimitusvastuun siirtämisestä välittäville solmuille sekä viestien kul-

jettamisesta liikkuvien solmujen mukana. Sama viesti voidaan lähettää verkkoon useita kertoja ja mahdollisesti myös rinnakkaisia polkuja pitkin, etenkin jos sen oletetaan parantavan viestin mahdollisuuksia päästä perille.

Viestien välitys verkossa on sitä tehokkaampaa, mitä paremman tilannekuvan valittu reititysmalli saa verkon olemassa olevista tai tulevista yhteyksistä. Jos mitään tietoa ei ole käytettävissä, joudutaan käyttämään viestien tulvitusta tai epideemistä reititystä niiden runsaasta resurssikulutuksesta huolimatta.

DTN-verkkojen epideemisen viestinvälityksen keventämiseksi on tutkittu myös erilaisia viestien redundanttiin koodaukseen perustuvia menetelmiä. Vastaanottaja pystyy rekonstruoimaan alkuperäisen viestin saatuaan riittävän määrän koodilohkoja. DTN-verkoissa sovellettaviksi tutkittuja koodauksia ovat poistokoodaus ja verkkokoodaus. Poistokoodauksessa viestin lähettäjä koodaa ja vastaanottaja purkaa, verkkokoodauksessa näitä kumpaakin tehdään jokaisella solmuvälillä erikseen.

Verkkokoodaus muuttaa niin perinpohjaisesti informaation välittämistä verkossa, että sen omaksuminen vienee aikaa. Toisaalta sen edut langattomissa verkoissa ovat kiistattomat ja esimerkkitoteutuksien COPE ja MORE kokemuksien mukaan myös helposti saavutettavat.

DTN-tekniikan vahvuudet ovat sen vika- ja viivesietoisuudessa. Kärjistetysti voidaan sanoa, että jos epideeminen DTN-reititys ei saa viestiä perille, ei siihen pysty mikään muukaan protokolla. Vielä lisää vikasietoisuutta saadaan yhdistämällä DTN-viestinvälitys erilaisiin koodaustekniikoihin.

Moni sovellusprotokolla vaatii päästä päähän -yhteyttä palvelimelle asti, vaikka itse sovelluksen toiminta etenkään käyttäjän kannalta ei sitä edellyttäisikään. Langattomien tekniikoiden ja liikkuvien käyttäjien yleistymisen vuoksi myös verkkosovellusten arkkitehtuuria halutaan uudistaa kestämään paremmin viiveitä ja häiriöitä. DTN-arkkitehtuurin viestinvaihtoon perustuva sovellusrajapinta on tapa toteuttaa sovelluksia, jotka tulevat toimeen epävarmoissa verkko-olosuhteissa.

Aktiivisesta tutkimuksesta huolimatta DTN-tekniikoilla ei vielä toteutettu aitoja palveluverkkoja. DTN-verkkojen tapaan toimiva United Villages -verkko yhdistää omalla DakNet-tekniikallaan jo satoja Aasian maaseutukyliä Internetiin bussien, jakeluautojen ja moottoripyörien kuljettamilla viestisolmuilla. Toiminta on laajenemassa voimakkaasti tulevina vuosina.

DTN-verkkojen tutkimuksessa pyritään kattamaan olemassa olevien verkkotekniikoiden jättämät aukot. Näin saadaan aikaan uusi OSI-mallia täydentävä verkkosovellusten arkkitehtuuri, jonka käyttämisestä hyvillä verkkoyhteyksillä ei ole mitään haittaa, mutta joka huonoissa olosuhteissa toimii ratkaisevasti nykyisiä tekniikoita paremmin. Tähän tavoitteeseen pääsemiseksi tarvitaan vielä runsaasti tutkimusta ja käytännön pilottisovelluksia, sillä DTN-verkoissa yhdistyvät usean haastavan sovellusalueen vaikeimmat ongelmat.

Lähteet

ACLY00 Ahlswede, R., Cai, N., Li, S. Y. R. ja Yeung, R. W., Network information flow. *Information Theory, IEEE Transactions on*, 46,4(2000), sivut 1204–1216.

AKG+07 Asokan, N., Kostiainen, K., Ginzboorg, P., Ott, J. ja Luo, C., Applicability of identity-based cryptography for Disruption-Tolerant Networking. *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*. ACM Press, 2007, sivut 52–56.

APS99 Allman, M., Paxson, V. ja Stevens, W., TCP Congestion Control, RFC2581, IETF, 1999.

AS06 Abdulla, M. ja Simon, R., A simulation analysis of multicasting in Delay Tolerant Networks. *WSC '06: Proceedings of the 38th conference on Winter simulation*. Winter Simulation Conference, 2006, sivut 2234–2241.

AY05 Akkaya, K. ja Younis, M. F., A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3,3(2005), sivut 325–349.

Bas07 Basu, I., DakNet: Internet Goes Rural Riding on a Motorbike, elokuu 2007. URL `http://www.govtech.com/dc/articles/128631`.

BBCD98 Blake, S., Black, D. L., Carlson, M. A. ja Davies, E., An Architecture for Differentiated Services, RFC2475, IETF, 1998.

BBL05 Burns, B., Brock, O. ja Levine, B. N., MV routing and capacity building in disruption tolerant networks. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, osa 1, 2005, sivut 398–408.

BCS94 Braden, R., Clark, C. ja Shenker, S., Integrated Services in the Internet Architecture: an Overview, RFC1633, IETF, 1994.

BDH+06 Brewer, E., Demmer, M., Ho, M., Honicky, R., Pal, J., Plauche, M. ja Surana, S., The challenges of technology research for developing regions. *IEEE Pervasive Computing*, 05,2(2006), sivut 15–23.

Bec07 Bech, M., Galathea3: Taking the NREN on an Expediton around the Globe, 2007. URL `http://tnc2007.terena.org/programme/presentations/show.php?pres_id=32`.

BEF+05 Brunner, M., Eggert, L., Fall, K., Ott, J. ja Wolf, L., Dagstuhl seminar on disruption tolerant networking. *SIGCOMM Computer Communications Review*, 35,3(2005), sivut 69–72.

BGJL06 Burgess, J., Gallagher, B., Jensen, D. ja Levine, B. N., MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. *INFOCOM 2006. 25th IEEE International Conference on Computer Communications*. IEEE, 2006.

BJS06 Burleigh, S., Jennings, E. ja Schoolcraft, J., Autonomous congestion control in Delay-Tolerant Networks. (2006). URL `http://hdl.handle.net/2014/39835`.

BLV07 Balasubramanian, A., Levine, B. ja Venkataramani, A., DTN routing as a resource allocation problem. *SIGCOMM Computer Communications Review*, 37,4(2007), sivut 373–384.

CBD02 Camp, T., Boleng, J. ja Davies, V., A survey of mobility models for ad hoc network research. *Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2,5(2002), sivut 483–502.

CBD+07 Cerf, V. G., Burleigh, S. C., Durst, R. C., Fall, K., Hooke, A. J., Scott, K. L., Torgerson, L. ja Weiss, H. S., Delay-Tolerant Networking Architecture, RFC4838, IETF, 2007.

CCS02 CCSDS, CCSDS File Delivery Protocol (CFDP), 2002. URL `http://public.ccsds.org/publications/archive/727x0b4.pdf`.

CDS74 Cerf, V., Dalal, Y. ja Sunshine, C., Specification of Internet Transmission Control Program, RFC675, IETF, 1974.

CFSD90 Case, J. D., Fedor, M., Schoffstall, M. L. ja Davin, C., Simple Network Management Protocol (SNMP), RFC1157, IETF, 1990.

CGG+04 Caldwell, D., Gilbert, A., Gottlieb, J., Greenberg, A., Hjalmtysson, G. ja Rexford, J., The cutting EDGE of IP router configuration. *SIGCOMM Computer Communication Review*, 34,1(2004), sivut 21–26.

CJKK07 Chachulski, S., Jennings, M., Katti, S. ja Katabi, D., Trading structure for randomness in wireless opportunistic routing. *SIGCOMM Computer Communications Review*, 37,4(2007), sivut 169–180.

CWJ03 Chou, P., Wu, Y. ja Jain, K., Practical network coding. *41st Allerton Conference on Communication, Control and Computing*, Monticello, IL, USA, lokakuu 2003, IEEE Computer Society, sivu 321.

CYS+06 Chen, L.-J., Yu, C.-H., Sun, T., Chen, Y.-C. ja Chu, H.-H., A hybrid routing approach for opportunistic networks. *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*. ACM Press, 2006, sivut 213–220.

DBF04 Demmer, M., Brewer, E., Fall, K., Jain, S., Ho, M. ja Patra, R., Implementing Delay-Tolerant Networking. Tekninen raportti, Intel Research Berkeley, 2004.

DFGV03 Dubois-Ferriere, H., Grossglauser, M. ja Vetterli, M., Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages. *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. ACM Press, 2003, sivut 257–266.

DFL01 Davis, J. A., Fagg, A. H. ja Levine, B. N., Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks. *Wearable Computers, 2001. Proceedings. Fifth International Symposium on*, 2001, sivut 141–148.

DGH+88 Demers, A., Greene, D., Houser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D. ja Terry, D., Epidemic algorithms for replicated database maintenance. *SIGOPS Operating Systems Review*, 22,1(1988), sivut 8–32.

DH98 Deering, S. ja Hinden, R., Internet Protocol, Version 6 (IPv6) Specification, RFC2460, IETF, 1998.

DMS04 Dingledine, R., Mathewson, N. ja Syverson, P., Tor: the second-generation onion router. *SSYM'04: Proceedings of the 13th conference on USENIX Security Symposium*. USENIX Association, 2004, sivut 21–21.

DO07 Demmer, M. ja Ott, J., Delay Tolerant Networking TCP Convergence Layer Protocol, IRTF, 2007. Work in progress (draft-irtf-dtnrg-tcp-clayer-00.txt).

DRCY00 Das, S. R., Robert Casta n. ja Yan, J., Simulation-based performance evaluation of routing protocols for mobile ad hoc networks. *Mobile Network Applications*, 5,3(2000), sivut 179–189.

DTNRG DTNRG, Delay-Tolerant Networking Research Group (DTNRG). URL `http://www.dtnrg.org/`.

DVBJ04 Dunkels, A., Voigt, T., Bergman, N. ja Jönsson, M., The Design and Implementation of an IP-based Sensor Network for Intrusion Monitoring. *Swedish National Computer Networking Workshop*, Karlstad, Sweden, marraskuu 2004.

EKM07 Effros, M., Koetter, R. ja Médard, M., Breaking network logjams. *Scientific American*, 296,6(2007), sivut 56–63.

Enn06 Enns, R., NETCONF configuration protocol, RFC4741, IETF, 2006.

Fall03 Fall, K., A delay-tolerant network architecture for challenged internets. *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2003, sivut 27–34.

Fall04 Fall, K., Custody transfer for reliable delivery in delay tolerant networks, IRTF, 2004. URL `http://www.dtnrg.org/papers/custody-xfer-tr.pdf`.

Fall06 Fall, K., Observations regarding a new (Internet) architecture, MSR Berkeley, 2006. URL `http://www.cs.berkeley.edu/~kfall/YANA.pdf`. Presented at MSR Cambridge.

FBW06 Fragouli, C., Boudec, J.-Y. L. ja Widmer, J., Network coding: an instant primer. *SIGCOMM Computer Communications Review*, 36,1(2006), sivut 63–68.

FC06 Farrell, S. ja Cahill, V., Security considerations in Space and Delay Tolerant Networks. *SMC-IT '06: Proceedings of the 2nd IEEE International Conference on Space Mission Challenges for Information Technology*. IEEE Computer Society, 2006, sivut 29–38.

FCG06 Farrell, S., Cahill, V., Geraghty, D., Humphreys, I. ja McDonald, P., When TCP Breaks: Delay- and Disruption- Tolerant Networking. *IEEE Internet Computing*, 10,4(2006), sivut 72–78.

FD06 Fall, K. ja Demmer, M., Disruption/Delay-Tolerant Networking Tutorial, 2006. URL `http://www.cs.berkeley.edu/~demmer/talks/dtn-tutorial-mobihoc-may06.pdf`.

FLZ+05 Fu, Z., Luo, H., Zerfos, P., Lu, S., Zhang, L. ja Gerla, M., The impact of multihop wireless channel on TCP performance. *IEEE Transactions on Mobile Computing*, 4,2(2005), sivut 209–221.

DAKNET FSM, United Villages India Brochure: DakNet. URL `http://www.firstmilesolutions.com/documents/United%20Villages%20India%20Brochure%20%20-%20English.pdf`.

FSW07 Farrell, S., Symington, S. ja Weiss, H., Delay-tolerant networking security overview, IRTF, 2007. Work in progress (draft-irtf-dtnrg-sec-overview-03.txt).

GD04 Grewal, J. ja DeDourek, J., Provision of QoS in wireless networks. *Second Annual Conference on Communication Networks and Services Research (CNSR'04)*. IEEE Computer Society, toukokuu 2004, sivut 337–340.

GGL07 Garetto, M., Giaccone, P. ja Leonardi, E., On the effectiveness of the 2-hop routing strategy in mobile ad hoc networks. *Communications, 2007. ICC '07. IEEE International Conference on*, 2007, sivut 3108–3113.

GMG+05 Gkantsidis, C., Miller, J., Goldberg, M., Rodriguez, P. ja Costa, M., Avalanche: File swarming with network coding, Microsoft, 2005. URL `http://research.microsoft.com/pablo/avalanche.aspx`.

Goth06 Goth, G., Delay-tolerant network technologies coming together. *IEEE Distributed Systems Online*, 7,8(2006), sivu 2.

GR05 Gkantsidis, C. ja Rodriguez, P. R., Network coding for large scale content distribution. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, osa 4, maaliskuu 2005, sivut 2235–2245.

GT02 Grossglauser, M. ja Tse, D. N. C., Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking*, 10,4(2002), sivut 477–486.

GXZ+06 Gong, Y., Xiong, Y., Zhang, Q., Zhang, Z., Wang, W. ja Xu, Z., WSN12-3: Anycast routing in delay tolerant networks. *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, marraskuu 2006, sivut 1–5.

HA06 Harras, K. A. ja Almeroth, K. C., Transport layer issues in delay tolerant mobile networks. *NETWORKING 2006 - Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems, 5th International IFIP-TC6 Networking Conference*, osa 3976 sarjasta *Lecture Notes in Computer Science*. Springer, 2006, sivut 463–475.

HABR05 Harras, K., Almeroth, K. ja Belding-Royer, E., Delay tolerant mobile networks (DTMNs): Controlled flooding schemes in sparse mobile networks. *NETWORKING 2005: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication Systems, 4th International IFIP-TC6 Networking Conference*, osa 3462 sarjasta *Lecture Notes in Computer Science*. Springer, 2005, sivut 1180–1192.

HAN05 Hanbali, A. A., Altman, E. ja Nain, P., A survey of TCP over ad hoc networks. *Communications Surveys & Tutorials, IEEE*, (2005), sivut 22–36.

HCS+05 Hui, P., Chaintreau, A., Scott, J., Gass, R., Crowcroft, J. ja Diot, C., Pocket switched networks and human mobility in conference environments. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 244–251.

HD03 Haas, Z. J. ja Deng, J., Dual Busy Tone Multiple Access (DBTMA). *Transactions on Communications*, 50,6(2003).

HKL+07 Hyyryläinen, T., Kärkkäinen, T., Luo, C., Jaspertas, V., Karvo, J. ja Ott, J., Opportunistic email distribution and access in challenged heterogeneous environments. *CHANTS '07: Proceedings of the second workshop on Challenged networks CHANTS*. ACM Press, 2007, sivut 97–100.

HM+06 Hart, J., , Martinez, K., Ong, R., Riddoch, A., Rose, K. C. ja Padhy, P., A wireless multi-sensor subglacial probe: design and preliminary results. *Journal of Glaciology*, 52,178(2006), sivut 389–397.

HYW+06 Heidemann, J., Ye, W., Wills, J., Syed, A. ja Li, Y., Research challenges and applications for underwater sensor networking. *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, osa 1, 2006, sivut 228–235.

DASM Hyyryläinen, T., DTN for Symbian Mobile Phones. URL `http://www.netlab.tkk.fi/u/thyyryla/dtn/`.

IEE99 IEEE Std 802.11-1999: Local and Metropolitan Area Networks, Part 11: Wireless LAN Medium Access Control and Physical Layer Specifications, IEEE, 1999.

IRTF IRTF, Internet Research Task Force (IRTF). URL `http://www.irtf.org`.

IPN ISOC, Interplanetary Internet Special Interest Group. URL `http://www.ipnsig.org/`.

JDPF05 Jain, S., Demmer, M., Patra, R. ja Fall, K., Using redundancy to cope with failures in a delay tolerant network. *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2005, sivut 109–120.

JFP04 Jain, S., Fall, K. ja Patra, R., Routing in a delay tolerant network. *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2004, sivut 145–158.

JLW05 Jones, E. P. C., Li, L. ja Ward, P. A. S., Practical routing in delay-tolerant networks. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 237–243.

JOW+02 Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L. S. ja Rubenstein, D., Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet. *Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*, osa 37. ACM Press, lokakuu 2002, sivut 96–107.

Ker07 Keränen, A., The ONE – Opportunistic Network Environment simulator, 2007. URL `http://www.netlab.tkk.fi/~akeranen/one/`.

KGK07 Katti, S., Gollakota, S. ja Katabi, D., Embracing wireless interference: analog network coding. *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2007, sivut 397–408.

KHA07 Kotz, D., Henderson, T. ja Abyzov, I., CRAWDAD data set dartmouth/campus (v. 2007-02-08), helmikuu 2007. URL `http://crawdad.cs.dartmouth.edu/dartmouth/campus`.

KKK06 Kim, M., Kotz, D. ja Kim, S., Extracting a mobility model from real user traces. *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, huhtikuu 2006, sivut 1–13.

KM07 Kiess, W. ja Mauve, M., A survey on real-world implementations of mobile ad-hoc networks. *Ad Hoc Networks*, 5,3(2007), sivut 324–339.

KRH+06 Katti, S., Rahul, H., Hu, W., Katabi, D., Médard, M. ja Crowcroft, J., XORs in the air: practical wireless network coding. *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM Press, 2006, sivut 243–254.

KZH07 Kate, A., Zaverucha, G. ja Hengartner, U., Anonymity and Security in Delay Tolerant Networks. *3rd International Conference on Security and Privacy in Communication Networks (SecureComm 2007)*, syyskuu 2007.

LDS03 Lindgren, A., Doria, A. ja Schelén, O., Probabilistic routing in intermittently connected networks. *SIGMOBILE Mobile Computing Communications Review*, 7,3(2003), sivut 19–20.

LFC05 Leguay, J., Friedman, T. ja Conan, V., DTN routing in a mobility pattern space. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 276–283.

LGE06 Lukac, M., Girod, L. ja Estrin, D., Disruption tolerant shell. *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*. ACM Press, 2006, sivut 189–196.

LJ06 Larsson, P. ja Johansson, N., Multi-user ARQ. *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, osa 4, 2006, sivut 2052–2057.

LLL07 Lin, Y., Liang, B. ja Li, B., Performance modeling of network coding in epidemic routing. *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*. ACM Press, 2007, sivut 67–74.

LLS+06 Leguay, J., Lindgren, A., Scott, J., Friedman, T., Crowcroft, J. ja Hui, P., CRAWDAD trace set upmc/content/imote (v. 2006-11-17), marraskuu 2006. URL `http://crawdad.cs.dartmouth.edu/upmc/content/imote`.

LR03 Li, Q. ja Rus, D., Communication in disconnected ad hoc networks using message relay. *Journal of Parallel and Distributed Computing*, 63,1(2003), sivut 75–86.

LSS07 Luby, M., Shokrollahi, A. ja Stockhammer, T., Raptor forward error correction scheme for object delivery, RFC5053, IETF, 2007.

LTZG06 Liao, Y., Tan, K., Zhang, Z. ja Gao, L., Estimation based erasure-coding routing in delay tolerant networks. *IWCMC '06: Proceedings of the 2006 international conference on Wireless communications and mobile computing*. ACM Press, 2006, sivut 557–562.

LWW06 Lin, F., Wang, Y. ja Wu, H., Testbed Implementation of Delay/Fault-Tolerant Mobile Sensor Network (DFT-MSN). *PERCOMW '06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*, Washington, DC, USA, 2006, IEEE Computer Society, sivu 321.

McD05 McDermott-Wells, P., What is Bluetooth? *Potentials, IEEE*, 23,5(2005), sivut 33–35.

MCP+02 Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R. ja Anderson, J., Wireless sensor networks for habitat monitoring. *WSNA '02: Proceedings of the*

1st ACM international workshop on Wireless sensor networks and applications. ACM Press, 2002, sivut 88–97.

MED07 McCarthy, B., Edwards, C. ja Dunmore, M., Network transparency in a mountain rescue domain, 2007. URL `http://tnc2007.terena.org/programme/presentations/show.php?pres_id=106`.

Mit04 Mitzenmacher, M., Digital fountains: a survey and look forward. *Proceedings of the IEEE Information Theory Workshop, 2004*. IEEE Press, lokakuu 2004, sivut 271–276.

MLS06 Mundur, P., Lee, S. ja Seligman, M., Routing in intermittent network topologies. *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*. ACM Press, 2006, sivut 385–389.

MN04 Mills, D. ja Nair, H., Timekeeping in the Interplanetary Internet, JPL/NASA, 2004. URL `http://www.eecis.udel.edu/~mills/ipin.html`.

MO06 Mukhtar, O. ja Ott, J., Backup and bypass: introducing DTN-based ad-hoc networking to mobile phones. *REALMAN '06: Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*. ACM, 2006, sivut 107–109.

Moc87 Mockapetris, P., Domain names - concepts and facilities, RFC1034, IETF, 1987.

MRPS04 Mahonen, P., Riihijarvi, J., Petrova, M. ja Shelby, Z., Hop-by-hop toward future mobile broadband IP. *IEEE Communications Magazine*, 42,3(2004), sivut 138–146.

OK04 Ott, J. ja Kutscher, D., Drive-thru Internet: IEEE 802.11b for automobile users. *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, osa 1, 2004.

OK06 Ott, J. ja Kutscher, D., Bundling the Web: HTTP over DTN. *Workshop on Networking in Public Transport (WNEPT 2006)*, Waterloo, Canada, elokuu 2006.

OKD06 Ott, J., Kutscher, D. ja Dwertmann, C., Integrating DTN and MANET routing. *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*. ACM Press, 2006, sivut 221–228.

OP07 Ott, J. ja Pitkänen, M. J., DTN-based Content Storage and Retrieval. *The First IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications: Proceedings of 7th IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM 2007)*, Helsinki, Finland, kesäkuu 2007, IEEE CS.

NETLAB Ott, J., Delay-Tolerant Networking at TKK Netlab. URL `http://www.netlab.hut.fi/~jo/dtn/`.

Ott07 Ott, J., Haastattelu sähköpostilla 22.11.2007. Liite 1.

Ott06 Ott, J., Application protocol design considerations for a mobile Internet. *MobiArch '06: Proceedings of first ACM/IEEE international workshop on Mobility in the evolving Internet architecture*. ACM Press, 2006, sivut 75–80.

Per01 Perkins, C. E. *Ad hoc networking*, sivut 1 – 28. Addison-Wesley Longman Publishing Co., Inc., 2001.

Per02 Perkins, C., IP Mobility Support for IPv4, RFC3220, IETF, 2002.

PFH04 Pentland, A. S., Fletcher, R. ja Hasson, A., DakNet: Rethinking Connectivity in Developing Nations. *Computer*, 37,1(2004), sivut 78–83.

Pos81 Postel, J., Internet Protocol, RFC791, IETF, 1981.

PWWM05 Petrova, M., Wu, L., Wellens, M. ja Mähönen, P., Hop of no return: Practical limitations of wireless multi-hop networking. *RealMAN*. IEEE, 2005.

RBF07 Ramadas, M., Burleigh, S. ja Farrell, S., Licklider Transmission Protocol - Specification, IRTF, 2007. Work in progress (draft-irtf-dtnrg-ltp-06.txt).

RHB+07 Ramanathan, R., Hansen, R., Basu, P., Rosales-Hain, R. ja Krishnan, R., Prioritized epidemic routing for opportunistic networks. *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*. ACM Press, 2007, sivut 62–66.

RKMF03 Ravot, S., Kelly, T. ja Martin-Flatin, J., TCP transfers over high latency/bandwidth networks & Grid DT. *Presented at Nordunet 2003*, elokuu 2003, URL `http://www.labunix.uqam.ca/~jpmf/talks/nordunet_20030826.pdf`.

SAP Sámi Network Connectivity project. URL `http://www.snc.sapmi.net/`.

SB07 Scott, K. ja Burleigh, S., Bundle Protocol Specification, RFC5050, IETF, 2007.

Sch05 Schultz, K., Stellar Steelhead WAN accelerator reduces network wait time, InfoWorld, marraskuu 2005. URL `http://www.infoworld.com/article/05/11/14/46TCriver_1.html`.

SFM06 Seligman, M., Fall, K. ja Mundur, P., Alternative custodians for congestion control in Delay Tolerant Networks. *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*. ACM Press, 2006, sivut 229–236.

SGP+07 Siekkinen, M., Goebel, V., Plagemann, T., Skevik, K.-A., Banfield, M. ja Brusic, I., Beyond the Future Internet–Requirements of Autonomic Networking Architectures to Address Long Term Future Networking Challenges. *FTDCS '07: Proceedings of the 11th IEEE International Workshop on Future Trends of Distributed Computing Systems*. IEEE Computer Society, 2007, sivut 89–98.

SH03 Small, T. ja Haas, Z. J., The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way). *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*. ACM Press, 2003, sivut 233–244.

Sha48 Shannon, C. E., A mathematical theory of communication. *Bell System Technical Journal*, 27(1948).

SK07 Song, L. ja Kotz, D. F., Evaluating opportunistic routing protocols with large realistic contact traces. *CHANTS '07: Proceedings of the second workshop on Challenged networks CHANTS*. ACM Press, 2007, sivut 35–42.

SKJH03 Song, L., Kotz, D., Jain, R. ja He, X., Evaluating location predictors with extensive Wi-Fi mobility data. *SIGMOBILE Mobile Computing Communications Review*, 7,4(2003), sivut 64–65.

SKZ06 Seth, A., Kroeker, D., Zaharia, M., Guo, S. ja Keshav, S., Low-cost communication for rural internet kiosks using mechanical backhaul. *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM Press, 2006, sivut 334–345.

SPH05 Sinha, R., Papadopoulos, C. ja Heidemann, J., Internet packet size distributions: Some observations, 2005. URL `http://netweb.usc.edu/~rsinha/pkt-sizes/`.

SPR04b Spyropoulos, T., Psounis, K. ja Raghavendra, C., Efficient routing in intermittently connected mobile networks: the multiple-copy case. *ACM/IEEE Transactions on Networking*, (2004).

SPR04a Spyropoulos, T., Psounis, K. ja Raghavendra, C., Efficient routing in intermittently connected mobile networks: the single-copy case. *ACM/IEEE Transactions on Networking*, (2004).

SPR05 Spyropoulos, T., Psounis, K. ja Raghavendra, C. S., Spray and wait: an efficient routing scheme for intermittently connected mobile networks. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 252–259.

Spy07 Spyropoulos, T., Course: Delay Tolerant Networking for Challenged Environments, 2007. URL `http://www-sop.inria.fr/planete/spyropoulos/course/syllabus.html`.

Ste Stewart, W., BITNET history. URL `http://www.livinginternet.com/u/ui_bitnet.htm`.

TAZ06 Tariq, M. M. B., Ammar, M. ja Zegura, E., Message ferry route design for sparse ad hoc networks with mobile nodes. *MobiHoc '06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*. ACM Press, 2006, sivut 37–48.

TZZ03 Tan, K., Zhang, Q. ja Zhu, W., Shortest path routing in partially connected ad hoc networks. *GLOBECOM '03*, osa 2, 2003, sivut 1038–1042 Vol.2.

VB00 Vahdat, A. ja Becker, D., Epidemic routing for partially connected ad hoc networks. Tekninen raportti, Duke University, 2000.

VSFA07 Vellambi, B. N., Subramanian, R., Fekri, F. ja Ammar, M., Reliable and efficient message delivery in delay tolerant networks using rateless codes. *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*. ACM Press, 2007, sivut 91–98.

War03 Warthman, F., Delay-Tolerant Networks (DTNS) - A Tutorial, IRTF, 2003. URL `http://www.dtnrg.org/docs/tutorials/warthman-1.1.pdf`.

WB05 Widmer, J. ja Boudec, J.-Y. L., Network coding for efficient communication in extreme networks. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant Networking*. ACM Press, 2005, sivut 284–291.

WDW07 Wang, Y., Dang, H. ja Wu, H., A survey on analytic studies of Delay-Tolerant Mobile Sensor Networks. *Wireless Communications and Mobile Computing*, 7,10(2007), sivut 1197–1208.

WEI+07 Wood, L., Eddy, W. M., Ivancic, W., McKim, J. ja Jackson, C., Saratoga: a delay-tolerant networking convergence layer with efficient link utilization. *Satellite and Space Communications, 2007 International Workshop on*, (2007).

WJMF05 Wang, Y., Jain, S., Martonosi, M. ja Fall, K., Erasure-coding based routing for opportunistic networks. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 229–236.

WPP+07 Wood, L., Peoples, C., Parr, G., Scotney, B. ja Moore, A., TCP's protocol radius: the distance where timers prevent communication. *Satellite and Space Communications, 2007 International Workshop on*, (2007).

WW06 Wang, Y. ja Wu, H., Replication-Based Efficient Data Delivery Scheme (RED) for Delay/Fault-Tolerant Mobile Sensor Network (DFT-MSN). *PERCOMW '06: Proceedings of the 4th annual IEEE international conference on Pervasive Computing and Communications Workshops*. IEEE Computer Society, 2006, sivu 485.

WW07 Wang, Y. ja Wu, H., Delay/Fault-Tolerant Mobile Sensor Network (DFT-MSN): A New Paradigm for Pervasive Information Gathering. *IEEE Transactions on Mobile Computing*, 6,9(2007), sivut 1021–1034.

XS02 Xu, S. ja Saadawi, T., Revealing the problems with 802.11 medium access control protocol in multi-hop wireless ad hoc networks. *Computer Networks*, 38,4(2002), sivut 531–548.

ZAZ05 Zhao, W., Ammar, M. ja Zegura, E., Multicasting in delay tolerant networks: semantic models and routing algorithms. *WDTN '05: Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*. ACM Press, 2005, sivut 268–275.

Zha06 Zhang, Z., Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges. *IEEE Communications Surveys*, IEEE, 2006.

ZKL+07 Zhang, X., Kurose, J. K., Levine, B. N., Towsley, D. ja Zhang, H., Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing. *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*. ACM Press, 2007, sivut 195–206.

ZNKT07 Zhang, X., Neglia, G., Kurose, J. ja Towsley, D., Performance modeling of epidemic routing. *Computer Networks*, 51,10(2007), sivut 2867–2891.

ZZ07 Zhang, Z. ja Zhang, Q., Delay/disruption tolerant mobile ad hoc networks: latest developments. *Wireless Communications and Mobile Computing*, (2007), sivut 1530–8669.

Liite 1. DTN-tutkijan haastattelu

Transcript of an e-mail interview with professor Jörg Ott [Ott07] follows. Each question is prefixed with a **Q** and each answer with a **A**. Like with any interview, the views given cannot be encompassing but only cover some aspects and cannot go much into depth.

0 Introduction.

Q 0.1 Having done some work on Quality of Service, I can think of DTN application architecture as a standard way of handling QoS issues at application level. Currently there is no standard way for an application of telling the network about different importance of network packets. DTN has it all in a tangible, simple way: three service classes, message expiry mechanism, selectable level of reception feedback (though some of Receipt-messages have been dropped from the standard for security reasons).

A 0.1 (which, of course, requires that the application knows what it wants :-) And then, there is not so much diversity with DTNs. Remember that we had some IP service class in the early days, none of which got ever really used.

Q 0.2 As a network manager, I'd like to have DTN capabilities implemented in every piece of networking hardware as a way to enable robust network mangement even without end-to-end connectivity. I'd love to say "reboot at once"or "revert to previous configuration"to a box I've just lost contact with but which can be reached by its immediate neighbors. With proper authentication, of course.

A 0.2 That's a nice one. I guess automated service/software updates to potentially fairly remote devices would clearly be interesting.

Q 0.3 DTN offers failure tolerancy beyond compare: it epidemic routing can not deliver a message to the recipient, then nothing else can either. One would like to think that this kind of robustness should be part of every protocol stack. For example, every wireless device could have a special dedicated emergency ad-hoc mode in which they convey official messages from rescue workers and authorities in a similar way that emergency number 112 is available even in a locked mobile phone.

A 0.3 Indeed, robustness – from various angles – is one of my favorites. I am not yet totally convinced by the rescue scenario though. My feeling is that we are lacking true experience. The rescue thing is good (in the sense that adding robustness is

good), but many applications – and particularly the users – may still not be happy with that. I have been in some discussions with user groups in that direction, and of the feedbacks was: there are occasion (when you get lost and feel threatened) you need to be able to push a button and hear a human voice immediately. Given the presumed need of rescue workers, an interesting question is, however, which portfolio of synchronous and asynchronous applications one should develop.

As a general thought: rescue teams are deployed and hence they are less ad-hoc than random mobile users. So, they will bring infrastructure even if the previous existing one has been deployed. The infrastructure may be mobile relays, portable GSM stuff, or just monstrous walkie-talkies. If this infrastructure can concentrate on attending to the most urgent needs – voice communications – and DTN can help free up wireless communication resources by using complementary short range technologies, then one can probably construct a scenario which approaches reality. This is just a guess, but I'd assume the future for such kind of operations will be some hybrid thing with DTN used for anything between fallback and mass volume data carriage.

1 DTN architecture

Q 1.1 How to do routing: try to live with epidemic routing or get fancy with predicting future contacts? PREP [RHB+07] and H-EC [CYS+06] appear to be simple and efficient.

A 1.1 I would say that erasure coding is primarily a mechanism for resource control which, as a side effect, may help message delivery rates.

We will not become happy with epidemic alone in large scale settings. But for campus or downtown areas with limited dimensions, some epidemic-style protocol (well: multi-copy routing) may be the best bet. As long as we have sufficiently large buffers compared to the message sizes, epidemic routing variants have the nice property of being simple. If the areas gets too large or the total offered load compared to the buffer size too large, we will need smarter approaches. Unfortunately, we still argue in terms of delivery probability (often well less than 100 %) and we measure delays in hours to days.

We will need to come up with something better in the long run if we want this technology to become usable in the real world.

Q 1.2 Network/erasure/rateless coding all look very promising, but will they be useful in

real applications? COPE [KRH+06] and similar opportunistic routing experiments reveal a huge performance and reliability increase for some traffic patterns.

A 1.2 These protocols are very depending on suitable traffic patterns and they rely on the results of the XORs being immediately useful. With opportunistic networking, we have rather rare encounters, often with quite some meeting time. XORs in the air, however, help if there is a lot of traffic and the wireless link really becomes the bottleneck. These may go together in some scenarios, but we will need to 1) make it work with the link layer (e.g., Bluetooth), 2) figure out how to map the respective roles to different devices (XORs in the Air uses APs, which we may not always be able to identify), and 3) we may have to know in advance how we should operate most efficiently. So, there is still some work to do to determine the actual applicability in the general case.

Q 1.3 Application <-> network interaction: Will DTN application model supersede layered OSI model for networked applications? Will DTN be the standard way of handling QoS at application level? Relaxing the network connection requirements makes life both easier and more complicated.

A 1.3 I don't think we should talk too much about QoS when we often have a hard time getting 75 % of the messages through. The entire notion of QoS will need to be revisited here. DTNs also target only a small set of applications (those being delay-tolerant) which means that we will still need a way to deal with the others.

However, I would hope that applications gain some more awareness of their operating environment so that they can adapt. We will maintain the concept of layers or some other functionally/logically structured blocks just help our minds by splitting complexity into manageable pieces. But maybe the structure will not be as strict.

Q 1.4 Application aware/application controlled routing: How to verify/authorize routing requests in Application-Hints header blocks? Running services other than http on tcp/80 is a common way to circumvent firewalls and proxies. Is there a way or, on the other hand, a need to protect DTN routing substrate from similar abuse?

A 1.4 People always find lots of "criminal energy"when motivated to abuse something (e.g., to obtain better service or any service in the first place). Port-80 tunneling being counteracted by means of deep packet inspection won't be suitable in mobile DTNs as sane application protocols (hopefully) will have security built in – which means an intermediary cannot figure out what the contents is about. We have been

investigating quite a lot, independent of the application-specific routing, how you can protect mobile infrastructure from abuse by individuals or groups. The answer is not clear yet, except that it is a hard problem – it is even hard to define "abuse"and some measure for it. A lot of the answers probably depend on the kind of DTN one is facing in a certain scenario; and this may also help defining proper means to validate whether the identification of application contents in a bundle is correct. One simple way may actually be uncertainty: if a user cannot say that using a certain "class"will provide some gain (and cannot really measure this either), then he cannot easily pick what he should to pretend to be.

2 DTN-powered applications

Q 2.1 Are different DTN-like applications too diverse to adopt a standard protocol? Or vice versa: are there things that cannot be described using DTN terminology, considering that the "store"part in "store-and-forward"can be made very short?

A 2.1 This is question on the applications probably boils down to two basic things: 1) Generality of a communication substrate traded off for application-specific support which is more tailored to the respective needs. We have this at the IP and at the transport layer of the Internet and, at the transport layer, applications happily choose what is available even though the fit may not be perfect. Continued suboptimality may lead to a protocol evolution (see SCTP and DCCP in the Internet world). 2) Constraints on the communicating entities: again, to take an example from the Internet, not everything is capable of running IP, nor is it necessary for everything to run IP. The current bundle protocol spec (don't forget to change your reference to RFC 5050) aims at generality and is relatively heavy if [all] you want to do is equip tiny sensors with DTN-style communication capabilities.

So, if the bundle protocol spec (or a successor) gets chosen as _the_ DTN protocol, I would expect to have a broad set of applications that can 'live with this' or find it a great fit and therefore choose 'buy' rather than 'making' their own. Besides this 'mainstream', there will always be environments for which the protocol just does not fit. And there we will have specific profiles and other protocols, maybe in complete isolation or using gatewaying of some sort.

Concerning the 'store' part: yes, it can be made indeed quite short and, in fact, packet-based communication has been referred to as store-and-forward networking because entire packets get stored before they are processed and forwarded (even though modern routers do not necessarily require this), in contrast to traditional

circuit switching dealing with a bit or an octet at a time. The key factors of interest will thus be bundle size, the resulting serialization delay, and the incurred (header, processing, etc.) overhead. This combination will decide whether shortening the storage time makes actually sense for a particular application. Given the bundle protocol spec, the overhead easily gets quite substantial if EIDs are long.

We should also not forget that applications interested in short delay are not necessarily delay-tolerant, thus raising the issue whether choosing a DTN protocol is a sensible decision in the first place.

Q 2.2 Which will be the first real-life applications? Mobile instant messaging? DakNet-like semi-transparent Internet extensions? Malware?

A 2.2 All of the above. Malware may come in only later as the bad guys have to find a sufficiently broad basis to grow on; so, early deployments should hopefully not be that badly affected (security is difficult enough anyway). And I would assume that local, serverless sharing applications are going to become relevant, regardless of whether they need DTNs or not, simply because people still pass USB sticks around despite 10+ years of ad-hoc networking.

Q 2.3 Showstoppers: With opportunistic routing and opportunistic application architecture, we've come a long way from traditional (deterministic) networking. DTN architecture is trying to do quite a many things in a radically new way. Is it trying to do too much? What are the greatest threats for DTN adoption?

A 2.3 Not finding the right applications. DTN is yet another tool and it can serve certain purposes but surely does not fit all. Even within DTNs, application scenarios are differing extremely in size, in stability, in predictability. If we apply the wrong approach (e.g., in routing) to some scenario it won't fly. If we create false expectations which we cannot fulfill, it won't either. 'Flying' in the sense of 'success' is difficult to define in the first place. It may range from adoption by some user community that finds it useful for a niche purpose to a huge market. So, I am afraid there is no single answer. Also, the lifetime of this technology is unclear: will the niches currently identified remain and maybe expand? Or will other technologies just solve the problems DTN is trying to address?

For instant messaging between mobile users, we currently have SMS and MMS. DTNs could help with inexpensive exchange of large amounts of data and higher data rates (shorter transfer times) if users are close by. But also wireless networking

infrastructures and pricing may solve (parts of) this problem. In contrast, underwater networking has limitations in range and rate, interplanetary communications has limitations in delay. These issues cannot as easily be addressed by alternative mechanisms.

So far, I think, DTN research has been humble in not trying to sell the universal solutions to all problems, which it clearly is not. Yet, the goals are ambitious but quite a bit stems from of observed (or perceived?) needs from some fields (e.g., ZebraNet). This is good because we have real users and real applications. We should avoid the pitfalls of MANET research (and thus we have to be very careful when talking about mobile users) and [not] build 100+ routing protocols without too many applications.

At the end of the day, users will decide about the success of a technology. Those may be marine scientist trying to get their meter reading in an affordable manner, not yet connected villages or nomads, or mobile phone users in industrialized countries. They all have in common that the technology must bring a perceived value add and just work. Everything else gets abandoned or replaced. Mobile phone users are probably the most demanding clientele, but there is also the biggest chance to create a community that builds applications on top of DTNs they find useful; and those may be applications we would never envision (who did imagine that downloading ringtones is going to be a serious business).

Q 3 Anything else You'd like to comment on?

A 3 Not much comes to my mind which would be written down easily. On the deployment side, a key issue will be to get]buy in from[some manufacturers building the right stuff into their equipment (particularly in the mobile user community) to get beyond this chicken and egg problem. (At least the applications are delay-tolerant, so you don't have to have suitable peers around you all the time.)